

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

**Využití virtuální instrumentace pro zpracování řečových signálů v
oblasti SMART technologií a Průmyslu 4.0**

**Virtual Instrumentation for Speech Signal Processing in SMART
Technology and Industry 4.0**

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

Zadání diplomové práce

Student: **Bc. Michal Skácel**

Studijní program: N2649 Elektrotechnika

Studijní obor: 2612T041 Řídicí a informační systémy

Téma: **Využití virtuální instrumentace pro zpracování řečových signálů
v oblasti SMART technologií a Průmyslu 4.0
Virtual Instrumentation for Speech Signal Processing in SMART
Technology and Industry 4.0**

Jazyk vypracování: čeština

Zásady pro vypracování:

Diplomová práce se věnuje problematice využití virtuální instrumentace pro zpracování řečových signálů (rozpoznávání řeči pro hlasové ovládání, filtrace rušení, apod.) v oblasti SMART technologií a Průmyslu 4.0. Cílem diplomové práce je návrh a realizace virtuálního prostředí pro simulaci hlasového ovládání včetně rušivého pozadí a filtrace v oblasti SMART technologií (hlasové ovládání inteligentní domácnosti, hlasové ovládání vozidla, apod.) a Průmyslu 4.0 (hlasové ovládání v rámci virtuální továrny, tj. volné ruce a volné oči).

Postup řešení:

1. Literární rešerše současného stavu problematiky zpracování řečových signálů v oblasti SMART technologií a Průmyslu 4.0 (rozpoznávání řeči pro hlasové ovládání, filtrace rušení, apod.). Klasifikace typů rušení, které se vyskytují v daných prostředích a jejich vliv na úspěšnost rozpoznávání řeči pro hlasové ovládání. Teoretický rozbor metod potlačení rušivého šumu pro rozpoznávací řeči.
2. Návrh a realizace virtuálního prostředí pro simulaci hlasového ovládání (implementace běžně dostupných rozpoznávačů) ve vývojovém prostředí LabVIEW. Vytvoření modelu reálného prostředí (např. výrobní linka, inteligentní domácnost, interiér automobilu, apod.) včetně vizualizace (ovládání a indukční prvky). Součástí modelu bude implementace běžně vyskytujících se rušení a jejich filtrace (implementace běžně využívaných metod) v daném prostředí (formou databáze), které degradují výslednou úspěšnost rozpoznávání. Navržený systém by měl být modulární a rozšiřitelný o další prvky (databáze rušení, další hlasové příkazy, apod.).
3. Tvorba databáze rušení vyskytujících se v oblasti SMART technologií a Průmyslu 4.0 pomocí měřicích mikrofónů G.R.A.S. 40PP a karty NI 9234 na bázi virtuální instrumentace pro potřeby realizovaného modelu virtuálního prostředí pro simulaci hlasového ovládání.
4. Návrh metodiky pro ověření funkčnosti navrženého modelu. Realizace série měření, které bude zkoumat vliv rušení a jejich následné filtrace na výslednou úspěšnost rozpoznávání.
5. Diskuze dosažených výsledků a zhodnocení účinnosti a využitelnosti aplikovaného řešení v oblasti SMART technologií a Průmyslu 4.0.

Seznam doporučené odborné literatury:

- [1] D'SOUZA, Charlton Rodney, et al. Voice Operated Control of a Motor Using LabVIEW. *Electrical and Electronic Engineering*, 2017, 7.2: 60-64. p-ISSN: 2162-9455/ e-ISSN: 2162-8459.
- [2] SHARMA, Ankush, Srinivas PERALA a Priva DARSHNI. Objects Control through Speech Recognition Using LabVIEW. *International Journal of Electronics and Computer Science Engineering*, 2013, V2N1-102-106. p.102-106. ISSN: 2333-9721.
- [3] GORECKY, Dominic, et al. Human-machine-interaction in the industry 4.0 era. In: *2014 12th IEEE International Conference on Industrial Informatics (INDIN)*. Porto Alegre, Brazil: IEEE, 2014. p. 289-294. DOI: 10.1109/INDIN.2014.6945523. Print ISSN: 1935-4576.
- [4] WITTASSEK, Tomáš. *Virtuální instrumentace I: učební text* [CD-ROM]. Ostrava: Vysoká škola báňská - Technická univerzita Ostrava, 2014. ISBN 978-80-248-3395-8.
- [5] UHLÍŘ, Jan. *Technologie hlasových komunikací*. Praha: Nakladatelství ČVUT, 2007. ISBN 978-80-01-03888-8.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **doc. Ing. Radek Martinek, Ph.D.**

Konzultant diplomové práce: Ing. Jan Baroš
Ing. Lukáš Šoustek

Datum zadání: 01.09.2018

Datum odevzdání: 30.04.2019



doc. Ing. Jiří Kozíorek, Ph.D.
vedoucí katedry



prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlášení

„Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“

V Ostravě dne 30. duben 2019

A handwritten signature in blue ink, appearing to read "Skica" followed by a stylized flourish.

Poděkování

Tímto bych rád poděkoval vedoucímu své diplomové práce, doc. Ing. Radku Martinkovi, Ph.D., za odbornou pomoc, podnětné rady a čas při vedení diplomové práce.

Abstrakt

Tato diplomová práce se zabývá automatickým rozpoznáváním řeči v oblasti Průmyslu 4.0 a SMART technologií pro následné testování vybraných filtračních metod. Nejprve se práce věnuje řešerši zabývající se uplatněním hlasového ovládání v konceptu Průmyslu 4.0 a SMART technologií. Dále se zabývá metodami automatického rozpoznávání řeči a filtračních metod. Primárně se tato práce zaměřuje na hlasové ovládání a filtraci rušení pomocí adaptivního algoritmu LMS a analýzy nezávislých komponent (ICA). V této práci byla realizována softwarová aplikace pro vytvoření databáze nahrávek rušení. Na základě těchto nahrávek byli realizovány tři vizualizace pro testování vybraných metod. Úspěšnost rozpoznání je vyhodnocena dle stavu rozpoznal/nerozpoznal, kdy každý příkaz byl 100x vysloven.

Klíčová slova

Automatické rozpoznávání řeči, Průmysl 4.0, SMART technologie, LabVIEW, slepá separace zdrojů, analýza nezávislých komponent, adaptivní algoritmus LMS.

Abstract

This thesis deals with automatic speech recognition in Industry 4.0 and SMART technology for subsequent testing of selected filtration methods. Firstly, the thesis deals with the search of voice control in the Industry 4.0 and SMART technologies. It also deals with methods of automatic speech recognition and filtering methods. Primarily, this work focuses on voice control and interference filtering using adaptive LMS and Independent Component Analysis (ICA). In this work, a software application was created to create a jam recording database. Based on these recordings, three visualizations were made to test selected methods. Recognition success is evaluated by state recognized / not recognized when each command was 100x pronounced.

Keywords

Automatic Speech Recognition, Industry 4.0, SMART Technology, LabVIEW, Blinds Source Separation, Independent Component Analysis, Adaptive LMS Algorithm.

Obsah

Seznam použitých zkratk	5
Seznam použitých symbolů	6
Seznam obrázků	7
Seznam tabulek	10
Úvod	11
1 Rešerše aplikačních oblastí rozpoznávání řeči	12
1.1 Uplatnění v rámci SMART technologií	12
1.1.1 Automatické rozpoznávání řeči v „inteligentních“ domech	12
1.1.1.1 Inteligentní řízení domu	12
1.1.1.2 Diktování	13
1.1.1.3 Velikost slovníku	13
1.1.1.4 Výkonnost	13
1.2 Uplatnění v rámci Průmyslu 4.0	14
1.2.1 Aplikace systémů ASR	14
1.2.1.1 Průmyslová robotika	14
1.2.1.2 Obrana a letectví	15
1.2.1.3 Telekomunikační průmysl	16
2 Koncept Průmyslu 4.0 a SMART technologií	17
2.1 Kyberneticko-fyzický systém	18
2.2 Průmyslový internet věcí (IoT)	18
2.3 Shrnutí principů Průmyslu 4.0	20
2.4 SMART technologie	21
2.4.1 „Inteligentní“ zařízení	21
2.4.1.1 Senzory	22
2.4.1.2 Akční členy	22
2.4.1.3 Řídící jednotka	22
2.4.1.4 Ovladač	22
3 Zpracování řečového signálu	23
3.1 Digitalizace	23
3.2 Parametrizace	23
3.3 Předzpracování	24

3.3.1	Segmentace.....	24
3.3.2	Ustředění	24
3.3.3	Preemfáze	25
3.3.4	Váhování oken.....	25
3.4	Metody krátkodobé analýzy	25
3.4.1	Analýza v časové oblasti	26
3.4.1.1	Krátkodobá energie	26
3.4.1.2	Krátkodobá intenzita	27
3.4.1.3	Krátkodobá funkce středního počtu průchodu signálu nulou.....	27
3.4.2	Analýza ve frekvenční oblasti	28
3.4.2.1	Krátkodobá Fourierova transformace.....	29
4	Automatické rozpoznávání řeči.....	30
4.1	Dynamické borcení času	30
4.2	Statistický přístup rozpoznávání řeči	31
4.2.1	Perceptuální lineární predikce	32
4.2.2	Metoda RASTA a její modifikace.....	32
4.2.3	Mel-frekvenční keprální analýza	33
4.2.4	Skryté Markovovy modely.....	34
4.2.5	Umělé neuronové sítě.....	35
5	Metody potlačování rušení	37
5.1	Jednokanálové metody	37
5.1.1	Wienerův filtr	38
5.1.2	Metoda spektrálního odečítání	38
5.1.3	Vlnková transformace	38
5.2	Vícekanálové metody	40
5.2.1	Metody adaptivní filtrace	40
5.2.1.1	LMS a RLS algoritmus	41
5.2.2	Metoda slepé separace zdrojů.....	42
5.2.2.1	Metoda nezávislých komponent.....	43
5.2.2.2	Předpoklady pro zpracování metodu ICA	43
5.2.3	Analýza hlavních komponent PCA	45
5.2.3.1	Rozklad pomocí singulárních čísel.....	45
5.2.4	Metoda tvarování paprsků.....	45

6	Popis zvukové karty Steinberg UR44 a mikrofonů RHODE NT5	47
6.1	Zvuková karta Steinberg UR44	47
6.2	Měřicí mikrofony RHODE NT5	48
6.2.1	Fantomové napájení	48
6.2.2	Kardioidní směrová charakteristika.....	49
7	Experimentální část	50
7.1	Software LABVIEW	50
7.1.1	Architektura aplikací	51
7.1.1.1	Komunikace mezi procesy	52
7.1.2	Zachytávání chyb a jejich řešení	52
7.1.3	Komunikace mezi LabVIEW a zvukovou kartou	54
7.1.3.1	Instalace.....	54
7.2	Tvorba databáze nahrávek rušení	56
7.2.1	Spuštění měření	57
7.2.2	Ukládání dat	58
7.2.3	Měření v polobezodrazové komoře	60
7.2.4	Měření v automobilech.....	61
7.2.5	Měření ve výrobní lince	63
7.2.6	Měření v letadle.....	65
7.3	Vizualizace	66
7.3.1	Převod řeči na text.....	66
7.3.2	Propojení pomocí virtuálního kabelu	67
7.3.3	Hlavní smyčka pro vyčítání dat.....	68
7.3.4	Vizualizace interiéru automobilu	69
7.3.4.1	Slovník příkazů	72
7.3.4.2	Ovládání aplikace	73
7.3.5	Vizualizace „inteligentního“ domu	73
7.3.5.1	Slovník příkazů	75
7.3.5.2	Ovládání aplikace	75
7.3.6	Vizualizace výrobní linky	76
7.3.6.1	Slovník příkazů	77
7.3.6.2	Ovládání aplikace	78
7.4	Vybrané metody filtrace a úspěšnost rozpoznání	79

7.4.1	Hledání optimálního nastavení parametrů algoritmu LMS	80
7.4.2	Analýza nezávislých komponent.....	81
7.4.3	Úspěšnost rozpoznání.....	81
7.4.3.1	Interiér automobilu	82
7.4.3.2	„Inteligentní domácnost“	87
7.4.3.3	Výrobní linka.....	91
Závěr		94
Použitá literatura		96
Příloha A		A

Seznam použitých zkratk

A/D	Analogově-digitální převod
ADALINE	Metoda využívající adaptivní lineární perceptron (Adaptive Linear Neuron)
ANN	Umělá neuronová síť (Artificial Neural Network)
ASR	Automatické rozpoznávání hlasu (Automatic Speech Recognition)
BF	Tvarovač paprsků (Beam Former)
BSS	Slepá separace zdrojů (Blinds Source Separation)
CC	Vyhovující třída (Class Compliant)
CPS	Kyberneticko-fyzikální systém (Cyber-Physical System)
CWT	Komplexní vlnková transformace (Complex Wavelet Transform)
DCT	Diskrétní kosinová transformace (Discrete Cosine Transform)
DNN	Hluboká neuronová síť (Deep Neural Network)
DTW	Dynamické borcení času (Dynamic Time Warping)
DWT	Diskrétní vlnková transformace (Discrete Wavelet Transform)
FFT	Rychlá Fourierova transformace (Fast Fourier Transform)
FIR	Konečná impulsní odpověď (Finite Impulse Response)
GMM	Model směsi Gaussových distribucí (Gaussian Mixture Model)
GUI	Grafické uživatelské rozhraní (Graphical User Interface)
HMI	Rozhraní člověk-stroj (Human Machine Interface)
HMM	Skryté Markovovy modely (Hidden Markov Models)
ICA	Metoda nezávislých komponent (Independent Component Analysis)
IIR	Nekonečná impulsní odpověď (Infinite Impulse Response)
IoT	Průmyslový internet věcí (Internet of Things)
LPC	Lineární prediktivní kódování (Linear Predictive coding)
MFCC	Mel-frekvenční keprstrální koeficienty (Mel-Frequency Cepstrum)
MLP	Vícevrstvý perceptron (Multilayer Perceptron)
PC	Osobní počítač (Personal Computer)
PCA	Metoda přímých komponent (Principal Component Analysis)
PLP	Perceptuální lineární predikce (Perceptual Linear Prediction)
SNR	Odstup signálu od šumu (Signal to noise Ratio)
SVD	Metoda singulárního rozkladu (Singular Value Decomposition)
SVM	Metoda podpůrných vektorů (Support Vector Machine)
VCD	definice hlasového příkazu (Voice Command Definiton)
VI	Vokální interakce, virtuální instrument (Vocal Interaction, Virutal Instrument)
WT	Vlnková transformace (Wavelet Transform)

Seznam použitých symbolů

A	Směšující matice
B	Bělicí matice
Bx(k)	Vybílený vektor
D	Diagonální matice
E_n	Krátkodobá intenzita
f	Frekvence [Hz]
f_{mel}	Frekvence v nelineární melovské škále [mel]
H	Separální matice
$H(k)$	Představuje k-tou komponentu Wienerova filtru
K(n)	Vektor zisků
Λ	Diagonální matice
O	Celkový vektor slova
P(n)	Inverzní autokorelační matice
$P(w_i O)$	Bayesova věta
Q_n	je krátkodobá charakteristika
$S(\omega, n)$	Krátkodobá Fourierova transformace
$s(k)$	značí vzorek akustického signálu
$s(n)$	n -tá hodnota rámce
$s'(n)$	ustředěná hodnota
V	Ortogonální matice sloupcových vektorů
W	Označení pro slovník
w(n)	Vektor koeficientů filtru
$w(n)$	Příslušný typ okénka
$X(m, n)$	Diskrétní vlnková transformace
x(n)	Vstupní vektor filtru.
$x[n]$	vstupní vzorek filtru
$X(\tau, s)$	Spojité Vlnková transformace
$Y(k)$	Spektrum rušeného signálu
$y[n]$	výstup filtru
Z_n	Krátkodobá funkce středního počtu průchodu signálu nulou
$\psi_{\tau,s}(t)$	Mateřská vlnka
$\Omega(\omega)$	Barkova frekvence [<i>bark</i>]
ω	Úhlová frekvence [$rad \cdot s^{-1}$]

Seznam obrázků

Obr. 1 Roční celosvětová výroba robotů [15]	15
Obr. 2 Jednoduchý model CPS	18
Obr. 3 Vrstvy internetu věcí	19
Obr. 4 Evoluce z vestavěných systémů ke kyberneticko-fyzickým systémům	20
Obr. 5 Postup předzpracování řečového signálu	24
Obr. 6 Krátkodobá energie příkazu "light on"	27
Obr. 7 Krátkodobá funkce středního počtu průchodu nulou pro příkaz „light on“	28
Obr. 8 Výsledek krátkodobé Fourierovy transformace pro příkazy "light on" a "light off"	29
Obr. 9 Principiální schéma DTW	30
Obr. 10 Hlavní blok rozdělení statistického rozpoznávání řeči	31
Obr. 11 Levo-pravý Markovův model o 6 stavech, převzato z [84]	35
Obr. 12 Principiální schéma jednokanálového systému	37
Obr. 13 Principiální schéma dvoukanálového systému	40
Obr. 14 Obecné blokové schéma adaptivního systému	41
Obr. 15 Základní model metody ICA	44
Obr. 16 Čelní panel zvukové karty Steinberg UR44 [79]	47
Obr. 17 Zadní panel zvukové karty Steinberg UR44 [79]	47
Obr. 18 Zvuková karta Steinberg UR44 [80]	48
Obr. 19 Směrová charakteristika mikrofону RHODE NT5 [81]	49
Obr. 20 Mikrofon RHODE NT5 [81]	49
Obr. 21 Pohled na uživatelské rozhraní vývojového prostředí LabVIEW	51
Obr. 22 Ukázka návrhového vzoru producent/konzument	51
Obr. 23 Vlevo data enqueue, vpravo data dequeue	52
Obr. 24 Vlevo jsou data zapsána, vpravo vyčtena	52
Obr. 25 Vyhodnocení a zaslání zprávy „ERROR“	53
Obr. 26 Hlavní smyčka zpracovávající stav ERROR	53
Obr. 27 Stav WAIT EXIT v hlavní smyčce	53
Obr. 28 Generování notifikace jedné z vedlejších smyček	54
Obr. 29 Korektní ukončení aplikace	54
Obr. 30 Funkce WaveIO_Open	55
Obr. 31 Funkce WaveIO_Start	55
Obr. 32 Funkce WaveIO_Record	55
Obr. 33 Funkce WaveIO_Stop	56
Obr. 34 WaveIO_Close	56
Obr. 35 Inicializace hlavní smyčky	56
Obr. 36 Diagram algoritmu po zmáčknutí tlačítka START	57
Obr. 37 Stav MEASURE v hlavní smyčce	57
Obr. 38 Vyčítání hodnot do grafů v grafické smyčce	58
Obr. 39 Diagram algoritmu pro ukládání dat	58
Obr. 40 Vykonání stavu LOG	59
Obr. 41 Vykonání stavu LOG_FILE	59
Obr. 42 Čelní panel aplikace pro tvorbu nahrávek rušení	59

Obr. 43 Umístění mikrofónů, reproduktoru a zvukové karty v polobezodrazové komoře po měření rušení.....	60
Obr. 44 Ukázka naměřených rušení v polobezodrazové komoře.....	61
Obr. 45 Realizace měření ve automobilech.....	62
Obr. 46 Ukázka průběhu rušení. Primární mikrofón (index 0) zavřená okna vlevo, otevřená okna vpravo.....	63
Obr. 47 Realizace měření ve výrobní lince	64
Obr. 48 Ukázka naměřených rušení laserové/magnetické svářečky a lisu.....	64
Obr. 49 Realizace měření v letadle Alto 912GT	65
Obr. 50 Ukázka naměřených rušení při různých rychlostech a výškách.....	66
Obr. 51 VI pro komunikaci s rozpoznávačem.....	67
Obr. 52 Nastavení virtuálního kanálu do výchozího stavu	67
Obr. 53 Hlavní smyčka pro kontinuální vyčítání dat	68
Obr. 54 Škálování hodnot ze zvukové karty	68
Obr. 55 Posouvání nahrávek v poli v subVI Rotace sumu.....	69
Obr. 56 Zjednodušené blokové schéma algoritmu interiéru automobilu	70
Obr. 57 Ukázka zpracování příkazu přes vygenerovaný User event.....	70
Obr. 58 Vykonání stavu UPDATE NOISE	71
Obr. 59 Vykonání stavu pro zasílání dat do virtuálního kanálu, reproduktoru a provedení vyhodnoceného příkazu.....	71
Obr. 60 Čelní panel aplikace interiéru automobilu	73
Obr. 61 Zjednodušené blokové schéma algoritmu „inteligentního“ domu	74
Obr. 62 Vyhodnocení souřadnic a jejich zápis do proměnných a klastru	74
Obr. 63 Čelní panel aplikace "inteligentní" domácnosti	76
Obr. 64 Ukázka aktivace lisu v pomocné smyčce.....	76
Obr. 65 Zjednodušené blokové schéma algoritmu výrobní linky	77
Obr. 66 Čelní panel aplikace výrobní linky	79
Obr. 67 Ukázka filtrace algoritmu LMS a RLS	79
Obr. 68 Měřicí řetězec filtrace pro algoritmus LMS.....	80
Obr. 69 Měřicí řetězec hybridní filtrace LMS a ICA	81
Obr. 70 Spektrogram odfiltrovaného slova "off"	82
Obr. 71 Výsledky rozpoznání pro automobil Škoda Felicie, 80 km/h	83
Obr. 72 Výsledky rozpoznání pro automobil Škoda Felicie, 100 km/h	84
Obr. 73 Výsledky rozpoznání pro automobil Škoda Felicie, 130 km/h	85
Obr. 74 Porovnání spektrogramů při zavřených oknech, příkaz „decline call“, 80 km/h	86
Obr. 75 Porovnání spektrogramů při zavřených oknech, příkaz „decline call“, 100 km/h	86
Obr. 76 Porovnání spektrogramů při zavřených oknech, příkaz „decline call“, 130 km/h	86
Obr. 77 Výsledky rozpoznání pro "inteligentní" domácnost	89
Obr. 78 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení pračky.....	90
Obr. 79 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení vysavače	90
Obr. 80 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení ventilátoru	90
Obr. 81 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení myčky.....	91
Obr. 82 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení televize	91
Obr. 83 Výsledky rozpoznání pro výrobní linku.....	92

Obr. 84 Porovnání spektrogramů, příkaz „homing“, rušení laserové svářečky	93
Obr. 85 Porovnání spektrogramů, příkaz „homing“, rušení magnetické svářečky	93
Obr. 86 Porovnání spektrogramů, příkaz „homing“, rušení lisu	93

Seznam tabulek

Tabulka 1 Počet pásem filtru úměrných na frekvenci	33
Tabulka 2 Porovnání výhod a nevýhod metod pro extrakci příznaků a akustického modelu	36
Tabulka 3 Porovnání výhod a nevýhod uvedených metod pro odstranění rušení	46
Tabulka 4 Specifikace zvukové karty Steinberg UR44.....	48
Tabulka 5 Specifikace mikrofону RHODE NT5	49
Tabulka 6 Slovník příkazů pro hlasové ovládání interiéru automobilu	72
Tabulka 7 Slovník příkazů pro hlasové ovládání "inteligentní" domácnosti	75
Tabulka 8 Slovník příkazů pro hlasové ovládání výrobní linky	78
Tabulka 9 Optimální nastavení parametrů pro algoritmus LMS, vizualizaci interiéru automobilu	80
Tabulka 10 Optimální nastavení parametrů pro algoritmus LMS, vizualizaci "inteligentní" domácnosti	80
Tabulka 11 Optimální nastavení parametrů pro algoritmus LMS, vizualizaci "výrobní linky	81
Tabulka 12 Nastavení parametrů funkčního bloku ICA	81
Tabulka 13 Výsledky úspěšnosti rozpoznání pro automobil Škoda Felicie, 80 km/h.....	82
Tabulka 14 Výsledky úspěšnosti rozpoznání pro automobil Škoda Felicie, 100 km/h.....	83
Tabulka 15 Výsledky úspěšnosti rozpoznání pro automobil Škoda Felicie, 130 km/h.....	84
Tabulka 16 Výsledky úspěšnosti rozpoznání pro rušení pračky	87
Tabulka 17 Výsledky úspěšnosti rozpoznání pro rušení vysavače	87
Tabulka 18 Výsledky úspěšnosti rozpoznání pro rušení ventilátoru.....	88
Tabulka 19 Výsledky úspěšnosti rozpoznání pro rušení myčky	88
Tabulka 20 Výsledky úspěšnosti rozpoznání pro rušení televize.....	89
Tabulka 21 Výsledky úspěšnosti rozpoznání pro rušení laserové/magnetické svářečky a lisu.....	92
Tabulka 22 Obsah přiloženého CD disku.....	A

Úvod

Komunikace prostřednictvím mluvené řeči je základní a nejpoužívanější způsob přenosu informace mezi lidmi. Výjimkou není ani odvětví výpočetní techniky, kde je snahou, aby se plnohodnotným partnerem člověka v mluvené řeči mohl stát i počítač. Tento cíl je sledován především proto, že takový způsob komunikace může být prospěšný a výrazně usnadnit člověku život.

Systémy hlasové komunikace jsou stále častěji nasazovány do průmyslové a společenské praxe. Ve většině aplikací je použitelnost podmíněna na úzkou oblast úkolů, tj. omezenost slovníku nebo předem stanovené příkazy, které mají být počítačem rozpoznány. Všeobecné využití tak nacházejí různé systémy ovládání strojů a zařízení hlasovými povely nebo automatický přepis diktátu. Tyto systémy jsou vhodné zejména v případech jsou-li oči i ruce člověka zaměstnány jinými činnostmi. Tato diplomová práce je zaměřena na automatické rozpoznávání řeči v oblasti Průmyslu 4.0 a SMART technologií pro následné testování vybraných filtračních metod.

Práce je rozvržena na teoretickou a experimentální část. Teoretická část má charakter literární rešerše a shrnuje současný stav problematiky automatického rozpoznávání řeči. Rešerše vychází z aktuálních poznatků vědy a výzkumu, především pak z odborných článků s impaktním faktorem. Primárně se tato diplomová práce zaměřuje na klasifikaci a popis metod pro rozpoznávání řeči a metod pro potlačení rušení.

Úvodní kapitola teoretické části je zaměřena na popis uplatnění automatického rozpoznávání řeči v rámci Průmyslu 4.0 a SMART technologií. V následující kapitole jsou stručně vysvětleny a popsány tyto koncepty. Kapitola věnovaná zpracování řečových signálů je rozdělena na dvě části, kdy je nejprve popsáno předzpracování, které zahrnuje segmentaci, ustředění, preemfázi a váhování oken. Takto předzpracovaný signál je dále možné zpracovat pomocí metod krátkodobé analýzy v časové nebo frekvenční oblasti. Cílem krátkodobé analýzy je především potlačení redundance v řečovém signálu. Čtvrtá kapitola se zabývá metodami rozpoznávání řeči, kdy nejstarší metodou je tzv. dynamické borcení času, určenou pro aplikace malého slovníku. Dále jsou popsány úspěšnější statistické metody, které lze klasifikovat do čtyřech hlavních částí. Poslední kapitola teoretické části se věnuje metodám pro potlačení rušení obsažených v řečovém signálu. Tyto metody jsou klasifikovány do dvou základních skupin: jednokanálové a vícekanálové metody.

Experimentální část této práce je v prvním úseku zaměřena na popis vybraného software a realizace architektury dle zvoleného návrhového vzoru. Následně je popsána aplikace pro vytvoření databáze rušení a jejich ukázek. Na základě těchto nahrávek jsou realizovány tři vizualizace pro hlasové ovládání, jimiž jsou: interiér automobilu, inteligentní domácnost a výrobní linka. Pro hlasové ovládání byl zvolen komerčně dostupný rozpoznávač ve Windows od firmy Microsoft. Pro zkvalitnění řečových signálů, které byly kontaminovány rušením, byl zvolen adaptivní algoritmus LMS a metoda nezávislých komponent ICA. Testovány byly příkazy obsažené v definovaném slovníku, kdy každý příkaz byl vysloven 100x a na základě stavu rozpoznal/nerozpoznal byla vyhodnocena úspěšnost. Následně jsou obě metody porovnány a je proveden závěr.

1 Rešerše aplikačních oblastí rozpoznávání řeči

1.1 Uplatnění v rámci SMART technologií

Mezi tzv. SMART technologie se dnes již řadí také koncept „inteligentních“ domů, jehož cílem je automatizace domácností. Automatizace domů je automatické nebo poloautomatické ovládání a sledování domácích spotřebičů a vlastností obytných domů, jako jsou dveře, brány nebo okna [1] [2]. Spotřebič nebo přístroj lze definovat jako „inteligentní“ pokud jsou jeho údaje přístupné všem [3]. Ve skutečnosti chytré spotřebiče často používají vznikající technologie a komunikační metody [4], které umožňují různé služby jak pro spotřebitele, tak pro výrobce. Inteligentní domy lze tedy definovat jako takové, které mají charakteristiky centrálního ovládání domovních zařízení, síťové schopnosti, interakce s uživateli prostřednictvím inteligentních rozhraní apod. Pro přirozenou interakci s uživateli je jedna z nejvíce uživatelsky přívětivých metod vokální interakce (VI), která odpovídá fyzickému prostředí inteligentního domu. Systém VI, ke kterému lze přistupovat například z garáže, koupelny, ložnice nebo kuchyně, vyžaduje distribuovanou sadu mikrofónů a reproduktory spolu s centralizovanou jednotkou zpracování [5].

1.1.1 Automatické rozpoznávání řeči v „inteligentních“ domech

Automatické rozpoznávání řeči (ASR) lze rozdělit do tří základních skupin [6]. První skupinou jsou izolované systémy rozpoznávání slov (každé slovo je mluveno s pauzou před a po promluvě, například v bankovních nebo letištních telefonních službách). Druhou skupinou jsou systémy malého slovníku pro příkazy a řízení aplikací, a poslední skupinu tvoří systémy rozsáhlých slovníků pro aplikace spojitě řeči. Z hlediska ASR je inteligentní domácí systém směsí druhé a třetí třídy, kde je možné diktovat mailové zprávy a používat gramaticky omezené příkazy pro řízení domácností, tzv. příkazová a řídicí (angl. Command and Control) slovní zásoba apod.

Z velké části můžeme klasifikovat systém ASR prostřednictvím vokální interakce ve dvou hlavních kategoriích: nejprve jsou specifické kontrolní aplikace, které tvoří podstatu inteligentních domovů (hlasové ovládání oken, spotřebičů atd.), a druhé jsou obecné vokální aplikace, které mohou být použity ve všech systémech ASR [5].

1.1.1.1 Inteligentní řízení domu

Použitím technologie počítačového rozpoznávání řeči lze navrhnout a vytvořit víceúčelový bezdrátový systém, který může vypnout a zapnout jakýkoli elektrický spotřebič pro domácnost v závislosti na hlasovém příkazu uživatele. Thoraya Obaid a kol. [7] navrhl hlasový bezdrátový systém pro seniory a osoby se zdravotním postižením. Navrhovaný systém má dvě hlavní součásti, jmenovitě systém rozpoznávání hlasu a bezdrátový systém. Software LabVIEW byl použit k implementaci systému rozpoznávání hlasu. Pro bezdrátovou komunikaci byly použity moduly ZigBee. Navrhovaný systém stačí „vyškolit“ pouze jednou, kdy na základě přijatých dat v bezdrátovém přijímači spojených se spotřebiči jsou prováděny požadované operace.

Další systém domácí automatizace navrhli Dhawan S. Thakurand a Aditi Sharma [8], který může být integrován jako samostatná přenosná jednotka a umožňuje bezdrátově ovládat světla, ventilátory, klimatizační jednotky, televizory, bezpečnostní kamery, elektronické dveře, počítačové systémy, audiovizuální zařízení, a zapnout nebo vypnout veškeré spotřebiče, které jsou zapojeny do elektrické sítě. Celkový systém je řízen z mikrofону, který je připojen k čipu rozpoznávání řeči HM 2007. Tento čip vysílá hlasové příkazy v binární sekvenci k mikrokontroléru. Jednotka základní stanice rozhoduje a vysílá příkazy do vzdálené stanice pomocí transceiveru ZigBee.

Obecně lze tyto systémy nazvat jako funkce příkazu a řízení. Většina těchto aplikací má malou velikost slovníku (0 až 50 slov), což odráží operace potřebné pro ovládání zařízení. Například příkazy pro ovládání světel mohou obsahovat "zapnuto", "vypnuto" a možná i několik slov, v závislosti na tom, jaké další operace jsou k dispozici. Obvykle existuje přímé mapování mezi slovem nebo frází a jeho sémantikou, tj. akcí nebo významem, který má být spojen se slovy. Složitější příkazy však lze spravovat prostřednictvím souboru alternativ, kde je slovní zásoba omezena. Nicméně při vzrůstajícím počtu alternativních formulací roste i náročnost úlohy. Pro zvládnutí těchto úloh je nutná gramatická syntaxe, která specifikuje dané slovo a frázi s jejich přípustnými kombinacemi (alternativami) [5].

1.1.1.2 Diktování

Diktování zahrnuje automatický překlad řeči do písemné podoby. Systémy určené pro diktování zahrnují velké slovní zásoby a v některých případech i aplikace zahrnující další odborné slovníky pro danou aplikaci [9]. Systémy specifické pro konkrétní doménu tak mohou zvýšit přesnost rozpoznání [5].

1.1.1.3 Velikost slovníku

Mnoho faktorů v systému ASR pro VI je možné regulovat. Například variabilita řeči má většinou omezené použití. Jazyková flexibilita může být omezena pomocí vhodného gramatického návrhu apod. Schopnost přesně rozpoznat zachycenou řeč, která byla omezena závisí především na velikosti slovníku a odstupu signálu od šumu (SNR).

Rozpoznávání tak může být zlepšeno nejprve omezením velikosti slovní zásoby a druhým zlepšením je poměr signálu k šumu. Velikost omezení slovníku v systémech VI, je postavena na gramatice. Omezení slovní zásoby, například zkrácením jednotlivých příkazů, může vést ke zlepšení rozpoznání [10] [11]. Podobně kvalita zachycené řeči ovlivňuje přesnost rozpoznání [12]. Reakce v reálném čase je také žádoucí charakteristika v mnoha případech.

1.1.1.4 Výkonnost

Výkonnost systému je ovlivněna třemi aspekty: rychlost rozpoznávání, požadavky na paměťovou velikost a přesnost rozpoznávání. Tyto aspekty jsou ve vzájemném konfliktu, protože je poměrně snadné zlepšit rychlost rozpoznávání, kdy jsou sníženy požadavky na paměť na úkor i snížení přesnosti rozpoznání [13]. Úkolem návrhu systému hlasového rozpoznávání je proto omezit velikost slovní zásoby co nejvíce v každém okamžiku konverzace. Aby bylo možné určit, jak má být slovní zásoba omezena, je užitečné ji vztahovat k velikosti slovníku daného k přesnosti rozpoznávání při

dané hladině rušení. Automatické systémy rozpoznávání řeči často používají konkrétní domény a specifické aplikace přizpůsobené ke zlepšení výkonu, ale velikost slovní zásoby je důležitá v jakémkoli obecném jazyce ASR, bez ohledu na použitou techniku při implementaci. Některé systémy byly navrženy od základů, aby umožnily zkoumání účinků omezení slovní zásoby, jako je systém Bellcore [11], který obsahuje až 1,5 milionu individuálních slov. Přesnost rozpoznávání lineárně poklesla při logaritmickém zvýšení velikosti adresáře [11].

1.2 Uplatnění v rámci Průmyslu 4.0

Současné trendy v rozvojových technologiích tvoří důležité předpoklady pro budoucnost, posilované včasným a produktivním využíváním technologií pro obohacení lidského života. Zpracování řečových signálů, které zahrnuje automatické rozpoznávání řeči, syntetickou řeč a zpracování přirozeného jazyka, začíná mít významný dopad na podnikání, průmysl a snadnou obsluhu osobních počítačů. Kromě toho usnadňuje hlubší pochopení složitých mechanismů fungování lidského mozku. Pokroky v technologii rozpoznávání řeči během posledních pěti desetiletí umožnily širokou škálu průmyslových aplikací. Přesto dnešní aplikace poskytují malý náhled na bohatou budoucnost pro technologii řeči a hlasových rozhraní, která nakonec nahradí klávesnice s mikrofony pro navrhování rozhraní člověk-stroj (HMI) pro snadný přístup ke stále inteligentnějším strojům. To také ukazuje, jak se schopnosti systémů rozpoznávání řeči v průmyslových aplikacích vyvíjejí v průběhu času, aby se ujal další generace hlasových služeb [14].

1.2.1 Aplikace systémů ASR

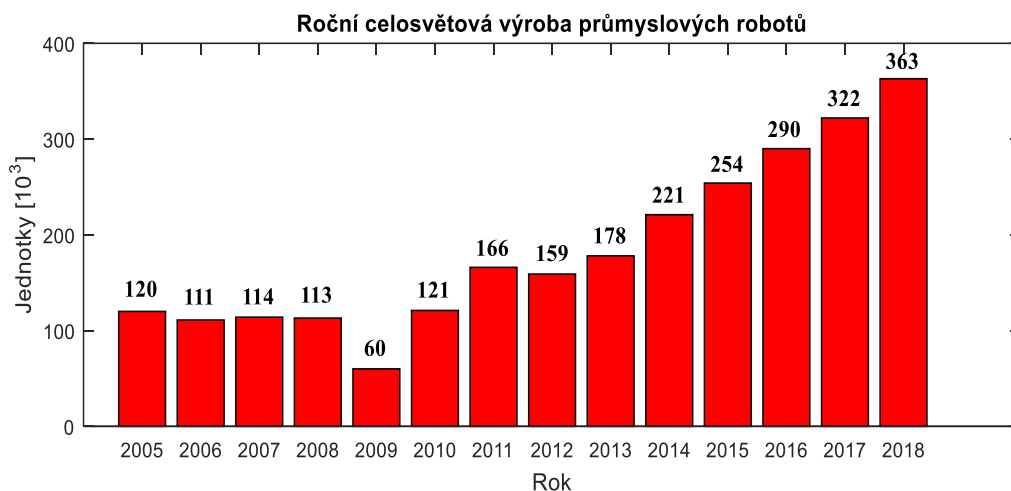
Jednou z motivací k výzkumu pro automatické rozpoznávání řeči spočívá v jeho průmyslové aplikaci, ve zjednodušení komunikace mezi lidmi a stroji. Omezením ručně psaných kroků vede ke schopnosti více komunikovat se zařízením, zatímco ruce nebo oči mohou provádět další úkoly. Další výhodou je, že vstupní signál řeči je mnohem rychlejší než jiné metody. Systémy automatického rozpoznávání řeči tak získávají uplatnění na poli průmyslového odvětví v různých sektorech, od obranného a zdravotnického až po spotřební. Vývoj rychlejších výpočetních schopností procesorů byl rovněž podporován v nákladově efektivních aplikacích systémů ASR, které jsou užitečné pro průmyslová odvětví. Systémy ASR aktuálně nacházejí širokou škálu aplikací v následujících oblastech [14]:

- Průmyslová robotika.
- Obrana a letectví.
- Telekomunikační průmysl.

1.2.1.1 Průmyslová robotika

Dnešní výkonné a levné mikroprocesory a zdokonalené algoritmy řídí komerční aplikace v oblasti počítačového příkazu, spotřebitele, zadávání dat, převod řeči do textu, telefonování a ověřování hlasu. Robustní rozpoznávací systémy pro ovládání a navigaci v osobních počítačích jsou nyní k dispozici [8]. Počítače s volným vstupem se používají v mnoha průmyslových aplikacích, kde jsou ruce a oči zaneprázdněny jinými úkoly, což umožňuje zadávání dat přímo do počítače bez klíčování nebo

přepisu. Technologie rozpoznávání řeči se úspěšně realizuje na průmyslových robotech a je široce přijímána v různých průmyslových odvětvích. Svět vyžaduje lepší produkty za nižší ceny s efektivními výrobními závody. Autonomie systémů, které vyžadují menší zásah obsluhy, by mohla napomoci k dosažení stejného cíle. Tento trend lze pozorovat na Obr. 1 v celosvětovém ročním přísunu robotů [14].



Obr. 1 Roční celosvětová výroba robotů [15]

A. Ragowski [16] vyvinul nový hlasový řídicí systém pro použití v robotizovaných výrobních buňkách spolu s nástroji umožňující jeho jednoduchou integraci do výroby. Jádrem celého systému je definice hlasového příkazu (VCD), který umožňuje snadný způsob definování sub-jazykového používání v jednotlivých aplikacích. Takto definované sub-jazyky mohou být charakterizovány velkou rozmanitostí možných formulací příkazu. Systém je sestaven z mikrofону, který je připojen do jednotky pro rozpoznání řeči (PC). Program je napsán na definici VCD v jazyce Arlang. Počítač pro rozpoznání řeči je propojen s řídicím počítačem, který obsluhuje robota, frézovací stroj a soustruh.

Další možností hlasového ovládání s využitím HMI lze použít v případech, kdy uživatelé se zdravotním postižením nejsou schopni ovládat své vlastní tělo na vozíku pomocí konvenčního joysticku a jsou často závislí na jiných osobách. Přidělením vozíku s novými rozhraními HMI je možné přispět ke společenské nezávislosti této skupiny uživatelů. V současné době se Robchair může pohybovat dynamicky prostředím v přítomnosti lidí, řízením jednoduchými nízko úrovněnými pokyny pro hlas nebo joystick [17].

1.2.1.2 Obrana a letectví

Aplikace rozpoznávání řeči v letectví se rychle vyvíjí a směřuje k návrhu budoucích letových palub. Nepřetržitý pokrok v oblasti počítačového hardwaru a softwaru využívá hlas pro řízení aplikací v pilotním prostoru, které z něj činí praktičtější, flexibilní a spolehlivý systém. V současné době několik vojenských aplikací používá ASR k poskytnutí dodatečných metod podpory rozhraní HMI aby se snížilo pracovní zatížení pilotů v oblasti pokročilého letectví. Boeing začleňuje ASR do nového letounu Strike Fighter. Euro Typhoon také začlenil tuto schopnost do svých letadel. Technologie rozpoznávání řeči je také použita v Joint Strike Fighter, který byl vyvinut Boeing a BAe Systems.

Kontinuální rozpoznání řeči je integrováno do kokpitu. Tento systém rozpoznávání řeči pomáhá pilotovi tím, že se vyhne rozptýlení způsobeném ručním ovládáním, přičemž se stále více zaměřuje na více kritické aspekty mise. Dodavatel systémů rozpoznávání řeči je pro toto letadlo ITT Industries Voxware (dříve VERBEX) [14] [18].

1.2.1.3 Telekomunikační průmysl

Svět telekomunikací, stejně jako rozpoznávání řeči se rychle mění a vyvíjí. Včasná aplikace technologie dosáhla různé stupně úspěchu. Rozpoznávání řeči bylo zavedeno do telekomunikační sítě na počátku 90. let ze dvou důvodů, jmenovitě snížení nákladů prostřednictvím automatizace obsluhovat funkce a poskytovat nové příjmy generování služeb, které byly dříve nepraktické kvůli souvisejícím nákladům na používání obsluhy. Telekomunikační služby vytvořené za účelem dosažení snížení nákladů zahrnují automatizaci služeb operátora, automatizace adresářové podpory, hlasové vytáčení atd.

Telekomunikační služby, které byly vytvořeny za účelem vytváření nových příjmů, zahrnují hlasové bankovní služby, hlasové volání, pomoc při telefonování, dokončení hovorů, informační služby, péči o zákazníky, integraci počítačů, telefonů a diktování hlasu. Navíc v komunikačních funkcích by hlasové ovládání mohlo být využíváno k naladění rádiových frekvencí voláním této frekvence.

Příslib do budoucna je výrazně vyšší výkon pro každou z těchto oblastí rozpoznávání řeči, kde budou implementovány robustnější algoritmy pro potlačování nežádoucího rušení, nové metody pro zpracování řečových signálů a kvalitnější hardware, což povede k robustním hlasovým rozhraním ke všem telekomunikačním službám [14] [19].

2 Koncept Průmyslu 4.0 a SMART technologií

Koncept Průmyslu 4.0 spočívá v přeměně výroby ze samostatných automatizovaných jednotek na plně automatizovaná a průběžně optimalizovaná výrobní prostředí. Takto vytvořená výrobní prostředí vyžadují vznik globální sítě, která je schopna propojovat výrobní zařízení do kyberneticko-fyzických systémů (CPS). CPS umožňuje samostatnou výměnu informací mezi systémy, vyvolání požadovaných akcí v závislosti na aktuálních podmínkách a vzájemné nezávislé kontroly. Stroje, senzory a IT systémy jsou vzájemně propojeny v rámci hodnotového řetězce. Díky propojení hodnotového řetězce a internetu jsou CPS schopny mezi sebou komunikovat, vzájemně reagovat a analyzovat data, aby mohly předvídat případné chyby nebo poruchy. Další vlastností těchto systémů je samostatná rekonfigurace a schopnost přizpůsobení se změnám podmínek v reálném čase.

V továrnách tak budou vznikat tzv. „inteligentní produkty“, které jsou jednoznačně identifikovatelné a lokalizovatelné. Tyto produkty budou umožňovat zaznamenat svou historii, aktuální stav a také cesty, které vedou ke vzniku finálního produktu.

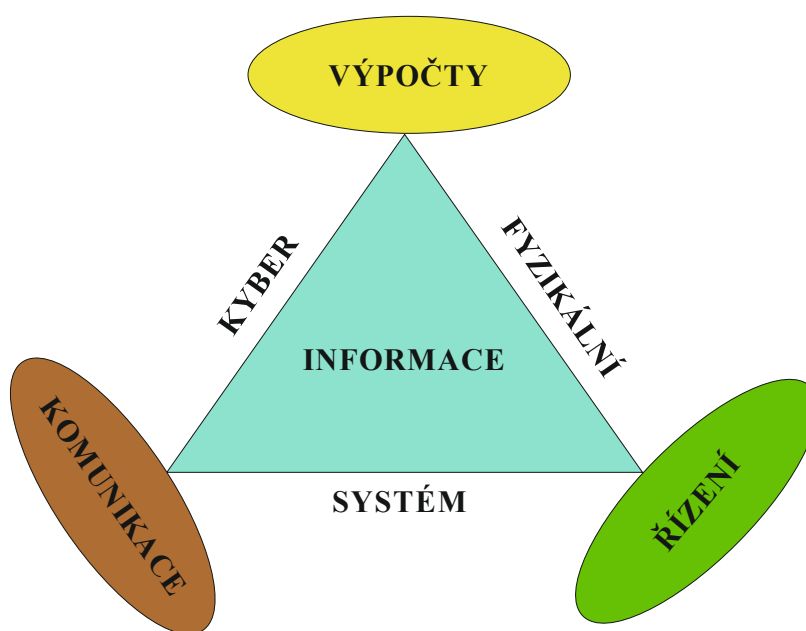
Základní charakteristiky inteligentních továren v rámci konceptu Průmysl 4.0 lze shrnout následovně [20]:

- Flexibilní výrobní procesy umožňují efektivní výrobu i malých výrobních dávek, které jsou přizpůsobeny individuálním požadavkům zákazníků.
- Fyzické prototypy jsou nahrazeny virtuálními návrhy výrobků, výrobních prostředků a výrobních procesů, jejich uvedení do provozu probíhá v rámci jednoho integrovaného procesu zapojujícího jak výrobce samotného, tak i jeho dodavatele.
- Výrobní zařízení se samo optimalizuje a konfiguruje v závislosti na parametrech zpracovávaného produktu.
- Izolované výrobní jednotky jsou nahrazeny plně automatizovanými a vzájemně propojenými výrobními linkami.
- Výrobní procesy jsou optimalizované v rámci hodnotového řetězce díky horizontálně i vertikálně integrovaným IT systémům.
- Roboti komunikující mezi sebou a výrobní zařízení činí do jisté míry autonomní rozhodnutí v reálném čase a tím zvyšují flexibilitu a efektivitu výrobního procesu.

2.1 Kyberneticko-fyzický systém

Kyberneticko-fyzický systém lze chápat jako spojení fyzických, síťových a výpočetních procesů. Tyto systémy jsou navrženy tak, aby docházelo k přehlednému integrování na sobě závislých fyzických součástí a výpočetních algoritmů. Vestavěný počítač s pomocí sítě kontroluje a řídí fyzické procesy. Využitím zpětné vazby mohou fyzické procesy ovlivňovat výpočty, a tím i samy sebe regulovat. CPS systémy spojují dynamiku fyzických procesů (například diferenciální rovnice) se softwarovými a síťovými procesy (programy a algoritmy).

Sociální a ekonomický potenciál těchto systémů je vysoký a celosvětově jsou vynakládány enormní investice podporující další vývoj této technologie [21].



Obr. 2 Jednoduchý model CPS

2.2 Průmyslový internet věcí (IoT)

Průmyslový internet věcí (IoT) je v informatice označení pro síť fyzických zařízení (vozidel, domácích spotřebičů), která jsou vybavena elektronikou, softwarem, senzory a síťovou konektivitou, která umožňuje se k těmto zařízením propojit a vyměňovat si tak vzájemně svá data. Přístroje IoT jsou například součástí širší koncepce domácí automatizace, známé také jako domotika. Velké inteligentní domácí systémy využívají hlavní router nebo switch, který uživatelům poskytuje centrální ovládání všech svých zařízení.

Jednou z klíčových aplikací inteligentní domácnosti je pomoc zdravotně postiženým a starším osobám. Tyto domácí systémy používají podpůrnou technologii pro přizpůsobení specifickým postižením člověka. Hlasové ovládání tak může pomoci například uživatelům s omezeními zraku a

pohyblivosti. K propojení se využívají běžně dostupné prostředky připojení jako je např. propojení kabelem, bezdrátové sítě, nebo data mobilních operátorů.

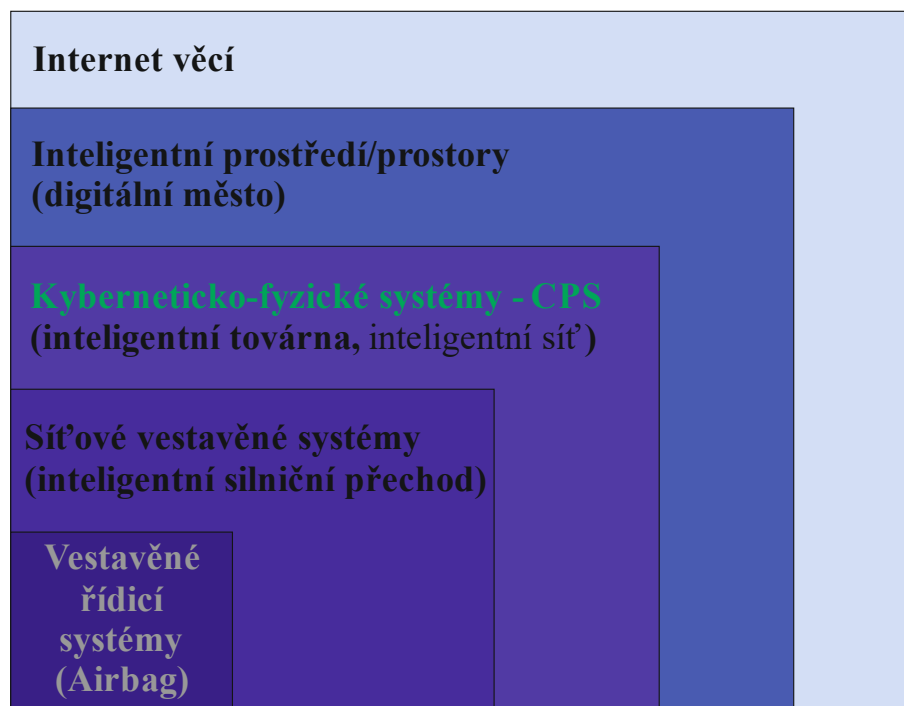
Internet věcí lze podle firmy Cisco rozdělit do vrstev, viz (Obr. 3). Jedná se o referenční model, který poskytuje jednotnou terminologii a objasňuje tok informací za účelem sjednocení pojmu internetu věcí. Nejnižší vrstvu tvoří fyzická vrstva, do které patří senzory, akční členy atd. Druhou vrstvu tvoří síťová komunikace mezi prvky a datovým centrem. Než se informace dostanou do datového centra, tak dochází k lokálním výpočtům. Další výpočty probíhají ve stacionárním datovém centru, který má formu velké databáze. Pátou vrstvu tvoří datová abstrakce. To znamená zaměření na užitečná data a přístup k nim. Na to navazuje vrstva aplikací, které už přímo pracují s daty a zpřístupňují je zákazníkům. Poslední je vrstva jsou procesy. Ta na základě výsledků z aplikací může spustit nový proces v jiných částech společnosti.



Obr. 3 Vrstvy internetu věcí

2.3 Shrnutí principů Průmyslu 4.0

- **Interoperabilita** – schopnost CPS komunikovat mezi sebou prostřednictvím internetu věcí/internetu služeb
- **Virtualizace** – realizace SMART Factory pro simulaci hlasového ovládání v rušeném prostředí v návaznosti na koncept Průmyslu 4.0
- **Decentralizace** – schopnost CPS jednat v rámci SMART Factory autonomně
- **Práce v reálném čase** – hlasové ovládání, kdy jsou získávána data a je tak možné zasahovat v reálném čase do výrobního procesu
- **Orientace na služby** – nabídka služeb CPS, obsluze, SMART Factory prostřednictvím internetu služeb
- **Modularita** – schopnost adaptace SMART Factory na změnu požadavků (změna rušení v továrním prostředí, tak aby se filtr přizpůsobil rušení a rozpoznávač řeči mohl správně interpretovat příchozí data.



Obr. 4 Evoluce z vestavěných systémů ke kyberneticko-fyzickým systémům

2.4 SMART technologie

V předešlé kapitole byl stručně popsán koncept Průmyslu 4.0, do kterého se řadí i IoT, což jsou běžná zařízení připojená k internetu za účelem sběru, výměny a zpracování dat.

Internet věcí mimo jiné zahrnuje také zařízení používaná v domácnostech. V dnešní době je již běžné pořídit si například inteligentní termostát s připojením k internetu a zapnout vytápění domu dlouho před příjezdem. Pro vyšší bezpečnost domova mohou posloužit IP kamery, které neustále sledují oblast před domem. Správným používáním IoT získá uživatel nejen vyšší komfort, ale také bezpečnost a možnost úspor.

S přibývajícím počtem zařízení připojených k internetu dochází ke stále větší poptávce po vhodném rozhraní, které uživatelům umožní jednoduše a intuitivně ovládat tato zařízení. Jedním z mnoha přístupů, které zvolili společnosti Amazon a Google jsou hlasem ovládaní inteligentní asistenti Amazon Echo a Google Home. Jedná se o samostatně stojící zařízení vybavené sadou mikrofónů a reproduktorů, kde veškerá komunikace s uživatelem probíhá pouze prostřednictvím hlasu, kdy je možné ovládat SMART televize, topení, ventilátory, světla a další zařízení. Mardiana a kol. [22] navrhli systém, který může hlasově ovládat různá zařízení. Uživatelské hlasové příkazy jsou převedeny do symbolů a jsou odeslány na server PC přes WI-FI síť. PC obsahuje aplikaci pro rozpoznávání hlasu vyvinutou na platformě Microsoft VisualBasic.net. Když je daný příkaz rozpoznán, tato informace je přenesena do řídicího obvodu přes paralelní port PC a odpovídající zařízení je zapnuto nebo vypnuto.

2.4.1 „Inteligentní“ zařízení

„Inteligentní“ zařízení (angl. Smart Device) představuje elektronické zařízení, které je vybaveno komunikačním rozhraním pro účely propojení, sdílení a interakce s dalšími inteligentními zařízeními, či uživatelem [23]. Zařízení tvořící inteligentní domácnost vzniká obvykle rozšířením funkcionality standardních zařízení použitých v domácnosti tak, aby bylo možné snadné monitorování a ovládání jejich funkcí uživatelem, ale také vzájemné propojení s dalšími inteligentními zařízeními. Mezi zařízení použitá v inteligentní domácnosti patří také řídicí jednotky a zařízení sloužící jako uživatelská rozhraní (ovládací panely, tablety) [24].

Mezi základní typy zařízení používaných v „inteligentní“ domácnosti patří:

- Senzory,
- akční členy,
- brány,
- řídicí jednotky a ovladače.

Existuje ale i celá řada komplexních zařízení, která v sobě kombinují dvě a více výše uvedených typů zařízení [25].

2.4.1.1 Senzory

V „inteligentních“ domech představují senzory prostředek, kterým jsou získávány informace o dění uvnitř domu, ale i v jeho okolí. Naměřená data ze senzorů následně umožňují uživatelům sledovat průběhy jednotlivých veličin díky kterým dostane „inteligentní“ domácnost zpětnou vazbu na akce provedené pomocí akčních členů. V oblasti „inteligentních“ domácnosti je široké spektrum zařízení, která zastávají funkci senzoru, jakou jsou například zařízení pro měření teploty, osvětlení nebo vlhkosti ovzduší. Dále existují senzory pro monitorování spotřeby vody, elektrické energie nebo plynu. Pro zajištění bezpečí domu a jeho okolí slouží například detektory pohybu, kouře nebo různých plynů v ovzduší.

2.4.1.2 Akční členy

Dalším prvek „inteligentní“ domácnosti jsou akční členy, které umožňují domácnosti interagovat s okolním světem. To znamená, že prostřednictvím „inteligentní“ domácnosti konají akce v reálném světě a ovlivňují tak své okolí. K tomu slouží různá elektronická (osvětlení, spínače) nebo mechanická zařízení (servomotory, pumpy) [25]. Mezi nejrozšířenější zařízení patří různé druhy stmívačů a vypínačů osvětlení, elektrické zásuvky umožňující na dálku zapínat a vypínat elektrická zařízení, nebo termostaty pro ovládání teploty. Další možností jsou ovladače žaluzií, rolet nebo garážových vrat.

2.4.1.3 Řídicí jednotka

Velmi častou součástí „inteligentní“ domácnosti je řídicí jednotka. Řídicí jednotka představuje počítač, který zpracovává data přicházející ze senzorů a s využitím předem definovaného chování vygeneruje odpovídající reakci, kterou následně vykoná akční člen. Řídicí jednotka se využívá nejen pro propojení senzorů a akčních členů, ale také jako brána pro připojení k internetu.

2.4.1.4 Ovladač

Ovladač umožňuje uživateli ovládat a monitorovat funkce „inteligentní“ domácnosti. Většina současných ovladačů je tvořena dotykovým displejem pro snadné a intuitivní ovládání, avšak s rozvojem automatického rozpoznávání řeči nejsou výjimkou ani ovladače určené pro hlasové ovládání „inteligentní“ domácnosti. Trendem dnešní doby je také ovládání pomocí chytrých telefonů a tabletů, kdy v kombinaci s mobilním internetem představují dříve nemyslitelné způsoby ovládání [25].

3 Zpracování řečového signálu

V předchozí kapitole byl nastíněn koncept Průmyslu 4.0, do kterého se řadí i pojmy jako chytrá továrna a chytrý dům. Pro chytrou továrnu je řečové rozpoznávání aplikováno s myšlenkou „rozšířené reality“, kdy obsluha ovládá svým hlasem příslušné stroje, a může tak kontrolovat dění kolem sebe. V odvětví „chytrých“ domů se jedná o udělování příkazů určených například pro rozsvícení světel, zatažení rolet nebo ovládání elektrických spotřebičů. Nicméně proto, aby byl příkaz úspěšně rozpoznán je zapotřebí vědět, jak se řečové signály zpracovávají a jak probíhá samotné rozpoznávání slov.

3.1 Digitalizace

Mluvená řeč obsahuje informace, které jsou obvykle získávány zpracováním zvukových kmitů řečového signálu snímaného mikrofonom [26]. Princip digitalizace spočívá v pravidelném odečítání hodnoty signálu pomocí analogově – digitálního A/D převodníku a jeho signálu v binární podobě. Tento proces zahrnuje provedení dvou kroků, a to vzorkování a kvantování s kódováním.

Vzorkování je proces diskretizace signálu v jeho časové oblasti, kdy jsou pravidelně odebírány vzorky v určených časových okamžicích. Tyto časové okamžiky jsou dány frekvencí označovanou jako vzorkovací frekvence, která musí splňovat Shannon-Kotělníkův teorém.

$$f_v = 2 \cdot f_{max}, \quad (3.1)$$

kde f_v je vzorkovací frekvence a f_{max} je maximální frekvence vyskytující se v signálu. Pro lidskou řeč se používá vzorkovací frekvence 8 kHz. Při nedodržení teorému vniká tzv. aliasing, který se projevuje jako slyšitelné zkreslení v nízkých frekvencích.

Kvantizace spočívá v zaokrouhlení hodnot signálu získaného při vzorkování na předem definované kvantizační hladiny. Mezi těmito hladinami se nachází rozhodovací stupně, které definují hodnotu kvantizačního stupně. Pokud bude například hodnota amplitudy signálu vyšší než rozhodovací stupeň, je kvantována pomocí vyššího kvantizačního stupně a naopak. S tímto procesem je spojeno kvantizační zkreslení, což je hodnota, o kterou jsme museli změnit původní amplitudu signálu [26].

Kódování je proces, při kterém se jednotlivým kvantizačním hladinám kvantovaného signálu přiřadí binární číslo.

3.2 Parametrizace

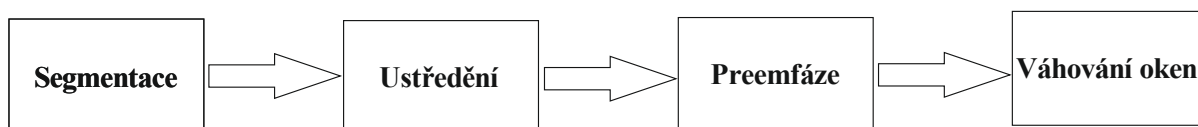
Úkolem parametrizace je potlačit nadbytečné informace řečového signálu. Tato nadbytečnost je způsobena tím, že každý člověk má originální hlasové ústrojí, a tedy i specifické vyslovování slov. Závisí to na způsobu artikulace, přízvuku, rychlosti řeči a dalších aspektech. Dokonce i stejný řečník může stejné slovo vyslovit jinak. Tyto rozdíly se projevují v délce jednotlivých úseků řeči a v intenzitě řečového signálu.

Průběh parametrizace můžeme rozdělit na:

- Předzpracování,
- Metody krátkodobé analýzy.

3.3 Předzpracování

Aby bylo možné využít metod krátkodobé analýzy pro parametrizaci signálu, je třeba signál předzpracovat. Předzpracování se skládá z následujících kroků, viz Obr. 5.



Obr. 5 Postup předzpracování řečového signálu

3.3.1 Segmentace

Pro zpracování řečových signálů metodami krátkodobé analýzy je nutné, aby měl signál stejné vlastnosti, tj. byl stacionární. Řečový signál je však považován za náhodný, a proto je nutné jej rozdělit na krátké časové úseky zvané segmenty. Volba délky těchto segmentů ovlivňuje výpočty všech metod krátkodobé analýzy a tím i výrazně celý proces segmentace. Délka segmentů musí být dostatečně malá, aby bylo možné postihnout i ty nejkratší fonémy, ale zároveň dostatečně velká, aby bylo možné postřehnout kvaziperiodický charakter řečového signálu. Nejčastěji se používá délka 10 - 30ms. Při segmentaci je možné, aby se sousední segmenty vzájemně překrývaly, což vede k lepšímu vyrovnaní průběhů počítaných parametrů signálu.

3.3.2 Ustředění

Jelikož může při snímání signálu dojít k jeho stejnosměrnému posunutí v kladném nebo záporném směru je nutné aplikovat ustředění. Posunutí může mít negativní dopad na další zpracování signálu, zejména na výpočty funkce krátkodobé energie. Ustředění jednoho segmentu lze vyjádřit následující rovnicí:

$$s'(n) = s(n) - \frac{1}{N} \sum_{i=1}^N s(i) \quad (3.2)$$

pro $1 \leq n \leq N$,

kde $s'(n)$ je ustředěná hodnota, $s(n)$ je n -tá hodnota rámce a N je počet vzorků obsažených v jednom segmentu.

3.3.3 Preemfáze

Při výstupu vlny z artikulačního ústrojí do volného prostoru dochází k útlumu intenzity zvuku pro vyšší frekvenční složky. Pro zvýraznění vyšších frekvenčních složek je signál filtrován filtrem FIR 1. řádu, který je definován rovnicí:

$$y[n] = x[n] - \alpha x[n - 1], \quad (3.3)$$

kde $y[n]$ je výstup filtru, $x[n]$ je vstupní vzorek filtru v čase n , který je zpožděn o vzorkovací periodu $x[n - 1]$ a koeficient preemfáze α , který nabývá hodnot 0,95 - 1.

3.3.4 Váhování oken

Při segmentaci signálů dochází k nežádoucímu prosakování ve spektru. Prosakování vzniká při nespojitostech signálu na okrajích segmentu. V jeho důsledku se pak ve spektru objevují frekvence, které nemají s původním signálem nic společného. Pro omezení tohoto vlivu je použitím tzv. okénko. Okénko je posloupnost vah pro prvky mikrosegmentu. Aplikace okénka na signál se provádí vynásobením odpovídající hodnoty rámce a okna:

$$y(n) = x(n)w(n), \quad (3.4)$$

kde $y(n)$ je výstupní posloupnost, $x(n)$ je vstupní posloupnost a $w(n)$ je aplikované okénko.

Nejběžněji používaný typ je Hammingovo okénko, které má optimální vlastnosti pro ASR. Popis Hammingova okna délky N vzorků je vyjádřeno rovnicí (3.5):

$$w(n) = \begin{cases} 0,54 - 0,46 \cos\left(\frac{2\pi n}{L-1}\right) & \text{pro } 0 \leq n \leq L-1 \\ 0 & \text{pro ostatní } n. \end{cases} \quad (3.5)$$

3.4 Metody krátkodobé analýzy

Úkolem krátkodobé analýzy je potlačit redundanci v řečovém signálu. Velikost dat je snížena tak, že signál pro další zpracování reprezentujeme souborem příznaků, které ze vzorků získáme, nikoliv však použitím vlastních nasnímaných vzorků. Snahou je, aby příznaky co nejlépe popisovaly vlastnosti signálu i jeho průběh. Metody krátkodobé analýzy se aplikují na jednotlivé segmenty získané při procesu segmentace. V každém segmentu je pak obsažen jeden příznak (skalár), anebo sada příznaků (vektor), který popisuje signál v daném segmentu.

Cílem krátkodobé analýzy je to, aby byl získaný popis jednoznačný a co nejméně ovlivněn okolními podmínkami. Musí tedy být co možná nejméně závislý na intenzitě signálu, na množství šumu v signálu a na řečníkovi. Metody krátkodobé analýzy je možné rozdělit na dvě kategorie. Analýza v časové oblasti pracuje s akustickým signálem jako s časovou posloupností navzorkovaných hodnot. Ve frekvenční oblasti je vnímán spíše spektrální charakter řečového signálů.

3.4.1 Analýza v časové oblasti

V časové oblasti jsou hodnoty zpracovány přímo z hodnot vzorků signálu. Příznaky, které získáme pomocí analýzy v časové oblasti, jsou většinou méně náročné na výpočet. Užitím jedné metody by pro mnohé fonémy hodnota příznaků zůstala stejná či se jen velmi málo změnila. Aby tedy bylo možné stanovit hranice všech fonémů, je vhodné použít více různých metod analýzy v časové oblasti a případně společně s nimi i metody pro analýzu ve frekvenční oblasti. Mezi tyto metody patří například krátkodobá energie, krátkodobá intenzita, počet průchodů nulou a krátkodobá autokorelační funkce. Většinu metod krátkodobé analýzy v časové oblasti lze vyjádřit rovnicí (3.6):

$$Q_n = \sum_{k=-\infty}^{\infty} \tau(s(k))w(n-k), \quad (3.6)$$

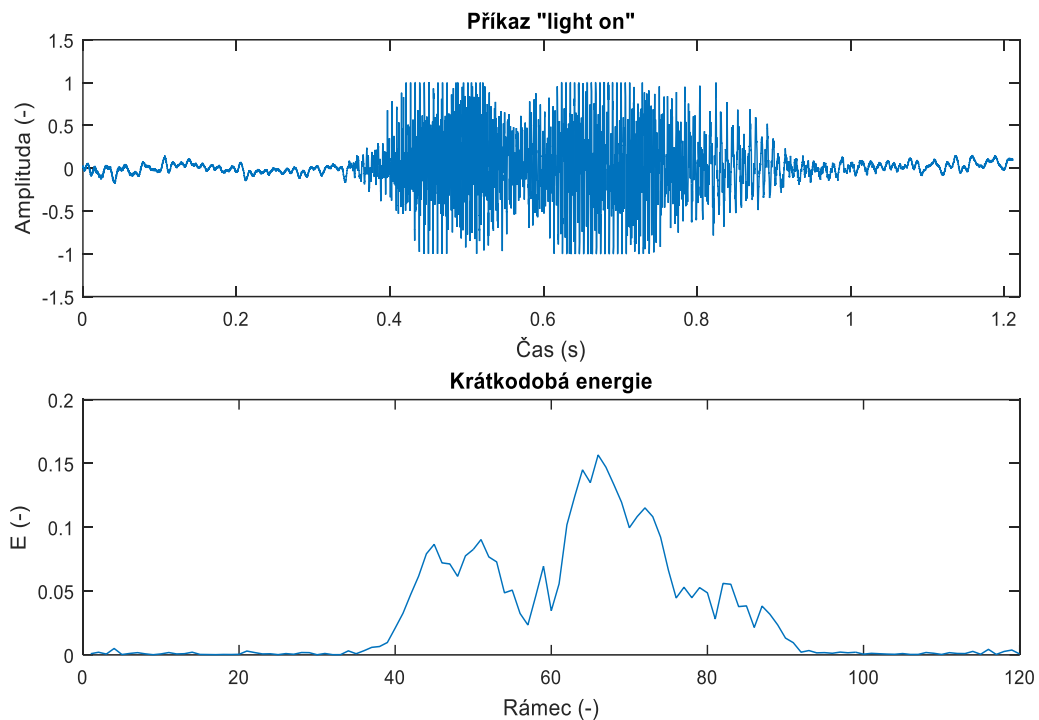
kde Q_n je krátkodobá charakteristika, $s(k)$ značí vzorek akustického signálu získaný pulsně šířkovou modulací v čase k , $\tau(\cdot)$ vyjadřuje příslušnou transformační funkci a $w(n)$ je váhová posloupnost, z které se vybírají, respektive váží vzorky $s(k)$ [26].

3.4.1.1 Krátkodobá energie

Tato metoda slouží pro výpočet energie segmentů. Hodnoty krátkodobé energie mohou být využity například při automatickém oddělování segmentů ticha od segmentů řeči nebo při oddělování znělých a neznělých částí promluvy [26]. Výpočet krátkodobé energie lze definovat vztahem (3.7):

$$E_n = \sum_{k=-\infty}^{\infty} [s(k)w(n-k)]^2, \quad (3.7)$$

kde $s(k)$ je vzorek signálu v čase k a $w(n)$ reprezentuje příslušný typ okénka. Při měření krátkodobé energie je vhodné volit délku mikrosegmentu shodnou s délkou okénka, a to o velikosti 10-20ms při frekvenci vzorkování odpovídající kvalitě zpracovávaného signálu.



Obr. 6 Krátkodobá energie příkazu "light on"

3.4.1.2 Krátkodobá intenzita

Krátkodobé intenzita, podobně jako u krátkodobé energie, slouží pro oddělování znělých a neznělých částí promluvy. Výhodou této metody je její „necitlivost“ na velké změny úrovně signálu. Funkci krátkodobé intenzity lze definovat vztahem (3.8).

$$E_n = \sum_{k=-\infty}^{\infty} s(k)w(n-k), \quad (3.8)$$

kde stejně jako u krátkodobé energie je $s(k)$ vzorek signálu v čase k a $w(n)$ reprezentuje příslušný typ okénka.

3.4.1.3 Krátkodobá funkce středního počtu průchodu signálu nulou

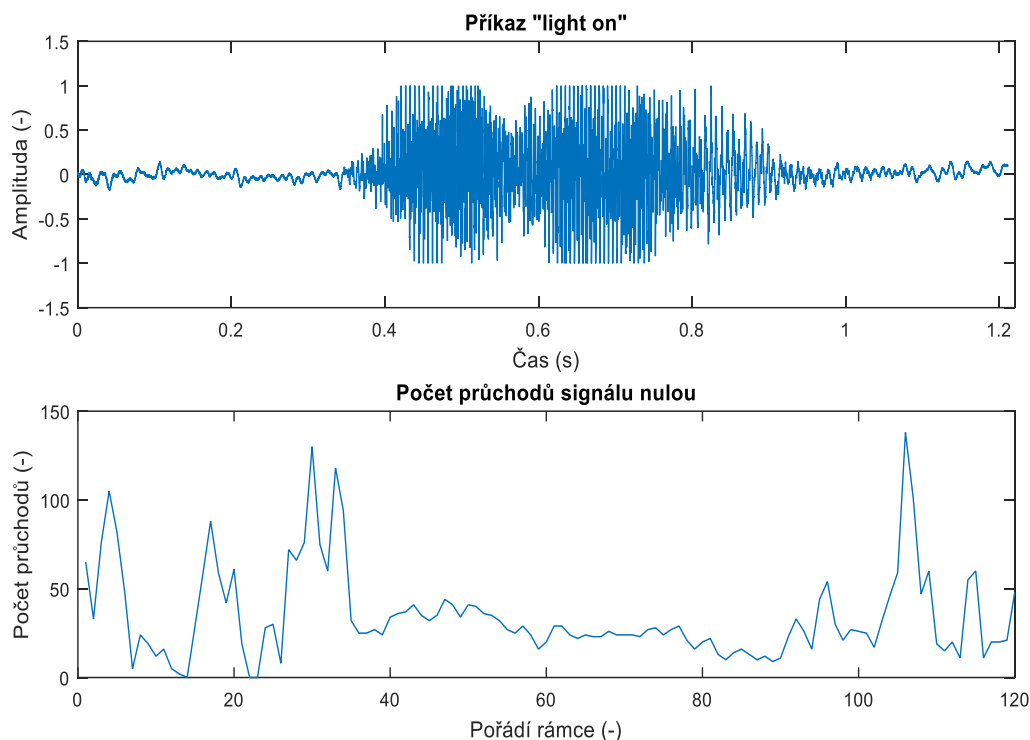
Na rozdíl od charakteristiky krátkodobé energie, zaměřené na sledování amplitudy signálu, poskytuje krátkodobá funkce středního počtu průchodu signálu nulou (Obr. 7) informaci o frekvenčních vlastnostech signálu. Krátkodobou funkci středního počtu průchodů nulou lze definovat rovnicí (3.9):

$$Z_n = \sum_{k=-\infty}^{\infty} |\text{sgn}[s(k)] - \text{sgn}[s(k-1)]|w(n-k), \quad (3.9)$$

kde

$$\text{sgn}[s(k)] = \begin{cases} 1 & \text{pro } s(k) \geq 0 \\ -1 & \text{pro } s(k) < 0, \end{cases}$$

a $w(n)$ je pravoúhlé okénko. Metody krátkodobé funkce středního počtu průchodu nulou jsou vhodné v procesech, kde se určují začátky a konce promluv pro určení základního hlasivkového tónu nebo zjištění přibližné hodnoty frekvence nejsilnějšího formantu. Nevýhodou těchto metod je nízká přenosová rychlost zpracování (přibližně 800 bit/s).



Obr. 7 Krátkodobá funkce středního počtu průchodu nulou pro příkaz „light on“

3.4.2 Analýza ve frekvenční oblasti

Mluvená řeč může být ve frekvenční oblasti reprezentována jako kompozice spektrální obálky charakterizující vlastnosti hlasového ústrojí a jemné struktury charakterizující buzení. Jak se mění buzení a tvar hlasového ústrojí, vytvářejí se rozdílné zvuky a mění se i spektrum signálu. Podobně jako u metod zpracování v časové oblasti, tak i ve frekvenční oblasti se pracuje s představou, že signál je přibližně stacionární. Nejčastěji používané postupy při analýze ve frekvenční oblasti jsou založeny na aplikaci krátkodobé Fourierovy transformace, viz Obr. 8 [26].

3.4.2.1 Krátkodobá Fourierova transformace

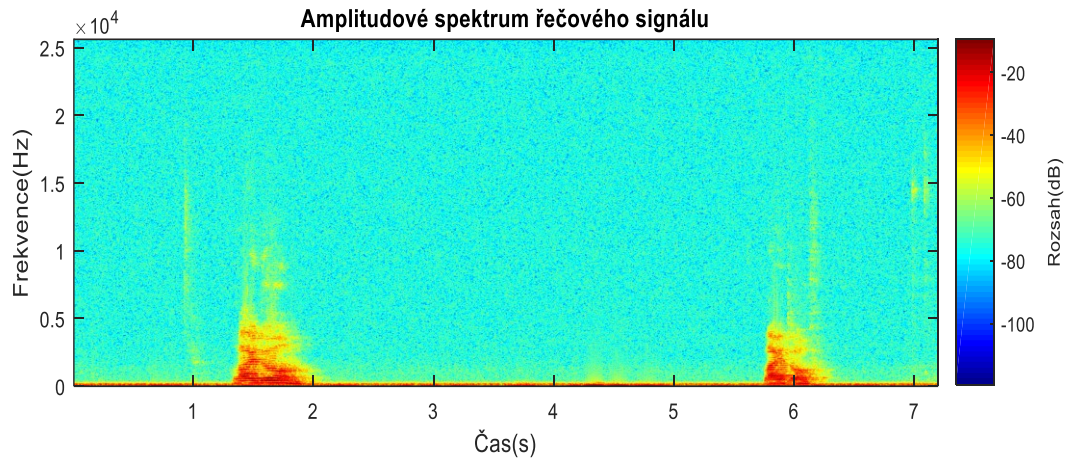
Krátkodobá Fourierova transformace $S(\omega, n)$ je definována vztahem (3.10)

$$S(\omega, n) = \sum_{k=-\infty}^{\infty} s(k)h(n-k)e^{-j\omega k}, \quad (3.10)$$

kde $h(n)$ je zatím blíže nspecifikovaná funkce okénka, která vybírá pro zpracování určený úsek signálu. Je zřejmé, že takto vyjádřený Fourierův obraz je funkcí spojitě proměnné frekvence ω , tak i diskrétní proměnné času n a odpovídá v podstatě konvoluci okénka $h(n)$ a $s(n)$ modulovaného $e^{-j\omega n}$ [21]. Po odvození dostaneme rovnici

$$S(\omega, n) = [s(n)e^{-j\omega n}] * h(n), \quad (3.11)$$

kde operace $*$ označuje konvoluci.



Obr. 8 Výsledek krátkodobé Fourierovy transformace pro příkazy "light on" a "light off"

4 Automatické rozpoznávání řeči

V předchozí kapitole byl popsán prvotní krok při rozpoznávání mluvené řeči, tedy analýza akustického signálu. Jak již bylo řečeno nejprve se provádí tzv. předzpracování a následně je vybrána některá z metod tzv. krátkodobé analýzy. Výsledek takto zpracovaného řečového signálu je dále předán algoritmu rozpoznávání řeči.

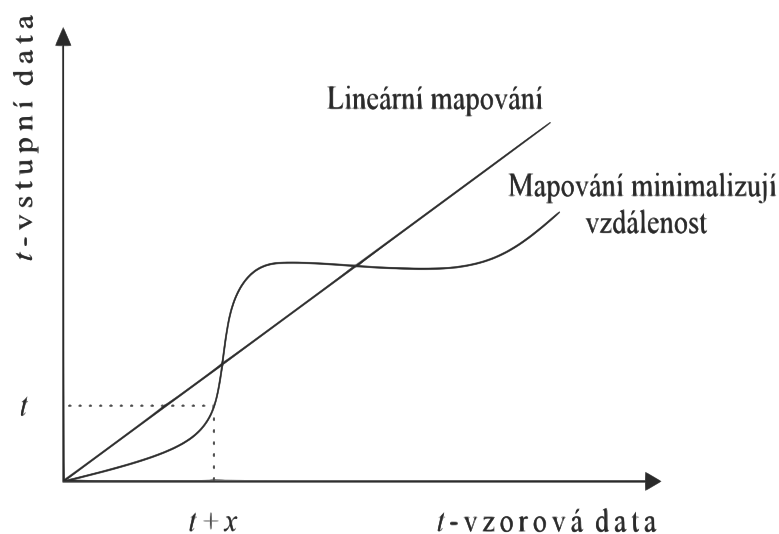
4.1 Dynamické borcení času

V oblasti rozpoznávání řeči se nejprve využívaly metody srovnání se slovem, do které se řadí i dynamické borcení času (DTW) [27].

Principem této metody je hledání minimální vzdálenosti mezi vstupními a vzorovými daty. Vzorová data představují například digitální nahrávky slov spolu s jejich přepisy, které tvoří slovník. Vstupní data představují nahrávky, které se mají rozpoznat. Pro počítání minimální vzdálenosti se používá tzv. dynamické programování, které je jedním ze způsobů řešení optimalizačních úloh a je založeno na Bellmanově principu optimality [28].

Algoritmus neporovnává pouze hodnotu vstupních dat v čase t s hodnotou vzorových dat v tomtéž čase, ale hledá takové mapování ze sekvence časových údajů ve vstupních datech do sekvence časových údajů ve vzorových datech, pro které je součet vzdáleností odpovídajících úseků dat minimální. Mapovací funkce nemusí být vždy lineární, podmínkou je pouze neklesající charakter. To znamená, že pořadí událostí v porovnávaných datech musí být shodné [29].

Příklad je naznačen na Obr. 9. Mapování může vypadat např. tak, že časy t_1, t_2 ve vstupních datech odpovídají po řadě časům $t_1 + 1s, t_2 - 1s$ s ve vzorových datech. Pro rozpoznávání řeči je tato vlastnost algoritmu výhodná, neboť umožňuje správně rozpoznávat i slova, která byla vyslovena jinou rychlostí než jejich uložený vzor. Rychlost se dokonce může průběžně měnit [29].



Obr. 9 Principiální schéma DTW

Nevýhodou metody DTW je její velká časová náročnost a také nutnost formulovat metriku pro výpočet vzdálenosti mezi daty, což může být někdy obtížné. Metoda se hodí pouze pro aplikace malého slovníku nebo rozpoznávání izolovaných číslic. Později byla nahrazena úspěšnějšími statistickými metodami.

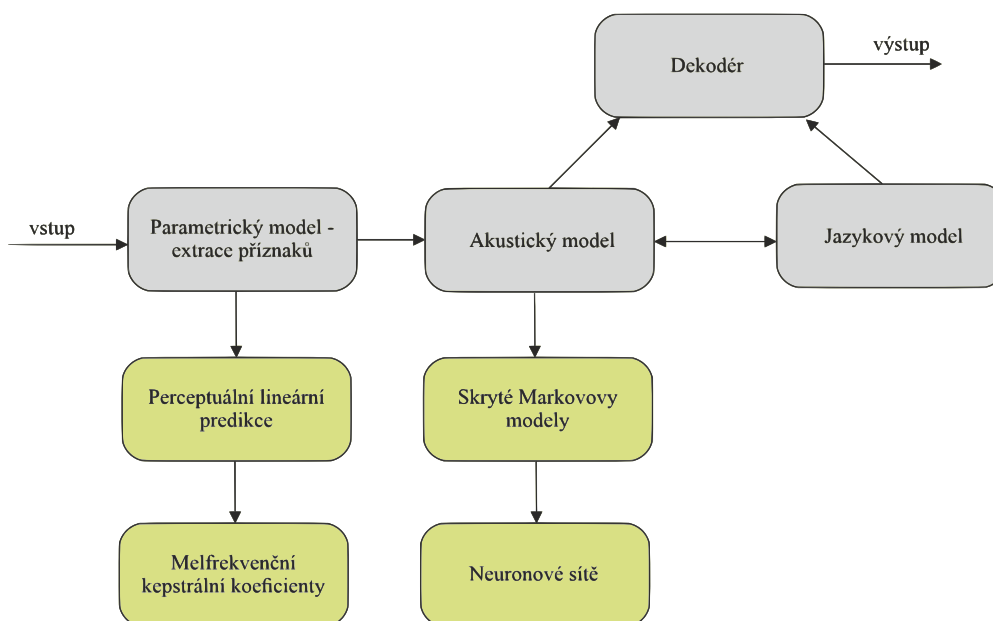
4.2 Statistický přístup rozpoznávání řeči

Systém využívající statistický přístup rozpoznávání řeči lze rozdělit do čtyř částí, viz Obr. 10. První částí je tzv. modul parametrizace nebo též extrakce příznaků, jehož úkolem je extrahovat ze vstupního řečového signálu takové příznaky, které co nejlépe zachovávají informaci o obsahu promluvy a zároveň účinně odstraňují značnou informační redundanci v řečovém signálu, viz kapitola 3. K tomuto účelu se v současné době využívají zejména metody perceptuální lineární predikce (PLP) a mel-frekvenční keprstrální koeficienty (MFCC) [30].

Druhou částí je akustický model, jehož úkolem je statisticky modelovat vztah mezi jednotlivými fonémy a jejich akustickou reprezentací. Akustický model hraje důležitou roli ve výkonu systému a je zodpovědný za výpočetní zátěž [31]. Tyto modely jsou převážně založeny na principu skrytých Markovových modelů (HMM) a umělých neuronových sítí (ANN) [30].

Úkolem jazykové modelu je co nepřesněji přiřadit pravděpodobnosti jednotlivým posloupnostem slova a tím pomoci akustickému modelu ve správném rozlišení mezi akusticky podobnými hypotézami o vyřčené promluvě. [32]. Obecně systémy rozpoznávání řeči používají *bi-gramové*, *tri-gramové* a *n-gramové* jazykové modely pro hledání správné slovní sekvence [33].

Posledním modulem je pak vlastní rozpoznávací algoritmus tzv. dekodér, který dokáže efektivně, nejlépe v reálném čase, vyhodnotit nejlepší z možných hypotéz [30].



Obr. 10 Hlavní blok rozdělení statistického rozpoznávání řeči

4.2.1 Perceptuální lineární predikce

Metoda PLP modeluje lidskou řeč na základě konceptu psychofyziky slyšení [34] [35], která obsahuje tři složky: kritické pásmo spektrální citlivosti, křivky stejné hlasitosti a vztah vyjadřující závislost mezi intenzitou zvuku a jeho vnímanou hlasitostí. Tato metoda vyřazuje irelevantní informace řečového signálu a zvyšuje tak rychlost rozpoznávání. PLP je identická s metodou lineárního prediktivního kódování (LPC) s tím rozdílem, že její spektrální charakteristiky byly transformovány tak, aby odpovídaly charakteristikám lidského sluchového ústrojí. Transformace se provádí vynásobením kritického pásma spektra rovnoměrným výkonem hlasitosti, který redukuje oblasti s vysokou a nízkou frekvencí a přivede je do středního frekvenčního pásma, kde je lidský sluch citlivější [36].

Výpočet PLP se provádí tak, že je řečový signál váhovan oknem a transformován do kmitočtové oblasti, obvykle pomocí rychlé Fourierovy transformace (FFT), kde jsou tyto vzorky převedeny na výkonové spektrum do barkovy stupnice pomocí aproximace:

$$\Omega(\omega) = 6 \cdot \ln \left(\frac{\omega}{1200\pi} + \sqrt{\left(\frac{\omega}{1200\pi} \right)^2 + 1} \right), \quad (4.1)$$

kde $\omega = 2\pi f$ je úhlová frekvence v rad/s a $\Omega(\omega)$ představuje barkovu frekvenci [bark].

Je známo, že vnímání tónů lidským uchem není lineárně závislé na frekvenci poslouchaného tónu. S měnící se frekvencí se mění vnímání zvuku, mimo jiné například subjektivní výška tónu. Tuto nelinearitu je možné zahrnout do procesu výpočtu parametrů, kdy je signál rozdělen do jednotlivých pásem, která odpovídají tzv. kritickým pásmům. V dalším zpracování signálu se poté pracuje pouze s energií v jednotlivých pásmech. Vzhledem k rozdílné šířce jednotlivých pásem je často tato energie normována k šířce pásma, aby nedocházelo k zvýšení vlivu širších pásem na vyšších frekvencích oproti pásmům užším. Při zpracování je nelineární filtrace realizována pomocí tzv. barkovy banky filtrů. V průběhu výpočtu parametrů je na signál možné aplikovat tzv. preemfázi a transformovat tak intenzitu zvuku na hlasitost. Tyto procesy způsobí potlačení dynamiky řeči a umožní tak lépe popsat signál výslednými koeficienty. Pro výpočet PLP keprstrálních koeficientů je použito LP analýzy. Tu lze získat po přechodu zpětnou Fourierovou transformací spektra do autokorelační oblasti. Následnou rekurzí jsou pak vypočteny hledané keprstrální koeficienty [37].

Nevýhodou této metody je závislost výsledných znakových vektorů, kterým nejlépe vyhovuje hluboká neuronová síť (DNN) [38].

4.2.2 Metoda RASTA a její modifikace

Metoda RASTA (z angl. **Rel**Ative **Spec**TrAl) byla navržena na základě vlastností lidského sluchu. RASTA je navržena tak, aby potlačovala spektrální složky, které se mění pomaleji nebo rychleji, než je typická rychlost změn řeči. Tato metoda byla prioritně konstruována jako nadstavba metody PLP pro aplikace s aditivním šumem (RASTA-PLP).

Zpracování probíhá podobně jako u metody PLP. Klíčovou myšlenkou metody RASTA-PLP je snaha potlačit konstantní nebo pomalu se měnící neřečové faktory signálu (obvykle rušení) v každé spektrální složce dříve, než budou vyčísleny parametry celo-pólového modelu [26].

Výhodou této metody je její jednoduchost, kdy při odstraňování vlivů zkreslení kanálu užívá relativně krátkou historii řádu stovek milisekund. Nevýhodou je zvýšení závislosti dat na jejich předchozím kontextu.

4.2.3 Mel-frekvenční keprstrální analýza

Mel-frekvenční spektrální analýza (MFCC), je nejpoužívanější metoda pro extrakci příznaků [39] [40]. Výpočet probíhá stejně jako u PLP s tím rozdílem, že při zpracování je nelineární filtrace realizována pomocí tzv. melovské banky filtrů. Melovské keprstrální koeficienty jsou získány zpětnou Fourierovou transformací logaritmu spektra. Vzhledem k reálnému a symetrickému výkonovému spektru je použito jednodušší diskrétní kosinové transformace (DCT). Výsledkem je sada koeficientů nazývaných akustické vektory [41]. Melovské keprstrální koeficienty se snaží „kompenzovat“ zejména nelineární vnímání frekvencí, a to využitím banky trojúhelníkových pásmových filtrů s lineárním rozložením frekvencí v tzv. melovské frekvenční škále, která je definována:

$$f_{mel} = 2595 \cdot \log_{10} \left(1 + \frac{f}{700} \right), \quad (4.2)$$

kde f_{mel} [mel] je odpovídající frekvence v nelineární melovské škále a f [Hz] je skutečná lineární frekvence obsažená v řeči [42].

Počet těchto filtrů obsažených v bance je úměrný vzorkovací frekvenci, viz Tabulka 1.

Tabulka 1 Počet pásem filtru úměrných na frekvenci

Vzorkovací frekvence [Hz]	Počet pásem filtru
8000	15
11000	17
16000	20
22000	22
44000	27

Trojúhelníkové filtry jsou standardně rozloženy přes celé frekvenční pásmo od nuly až do Nyquistovy frekvence. V některých úlohách je možné omezit přenášené pásmo, zvláště pak v případech, kdy je zbytečné umísťovat filtry do frekvenčních oblastí, ve kterých není užitečná energie signálu [26].

4.2.4 Skryté Markovovy modely

Skryté Markovovy modely (HMM) byly nejprve aplikovány na problematiku rozpoznávání izolovaných slov, tj. odděleně vyslovovaných slov z pevně daného slovníku omezené velikosti. Později bylo jejich použití rozšířeno na rozpoznávání souvislé řeči. HMM je model stochastického procesu, na něhož je možné pohlížet jako na pravděpodobnostní konečný automat, který v diskrétních časových okamžicích generuje náhodnou posloupnost vektorů pozorování [26] [43]:

$$O = o_1, o_2, o_3, \dots o_T, \quad (4.3)$$

kde O je celkový vektor slova, o_1 až o_T jsou vektory pozorování a T značí celkový počet pozorování a délku slova. Rozpoznávací algoritmus musí vybrat ze slovníku slovo w_i , které je s největší pravděpodobností nejbližší zkoumanému slovu O . Tuto pravděpodobnost lze vyjádřit jako:

$$\operatorname{argmax}_i P(w_i|O). \quad (4.4)$$

Pro slovník platí, že:

$$W = w_1, w_2 \dots w_n, \quad (4.5)$$

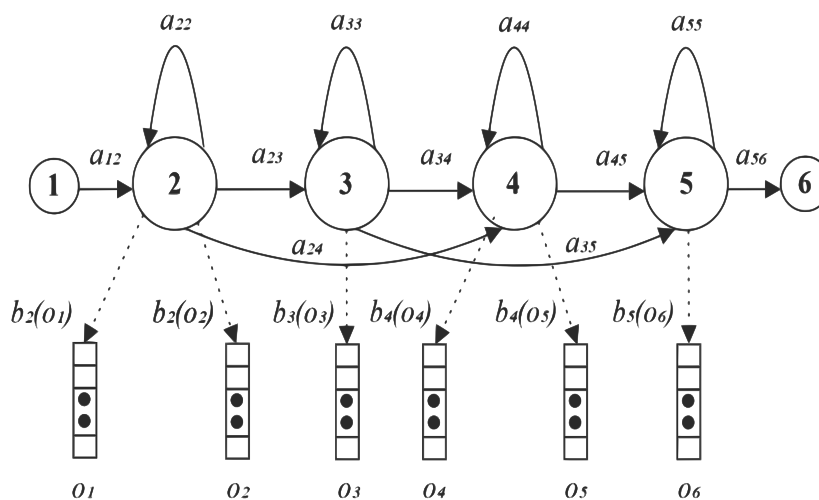
kde W je označení slovníku a w_1 až w_n jsou vzorová slova a n je jejich celkový počet. Tuto pravděpodobnost nelze vypočítat přímo, a proto je aplikována Bayesova věta [26]:

$$P(w_i|O) = \frac{P(O|w_i)P(w_i)}{P(O)} \quad (4.6)$$

Pravděpodobnosti $P(w_i)$ vyslovení jednotlivých slov ze slovníku bývají předem dány. Z výše uvedeného vztahu tedy vyplývá, že pro určení slova, které bylo vysloveno s největší pravděpodobností, potřebujeme spočítat pouze pravděpodobnosti $P(O|w_i)$. Pozorování O je však vektor a přímý výpočet sdružené podmíněné pravděpodobnosti $O = (o_1, o_2, o_3, \dots o_T|w_i)$ není možný. Místo toho lze využít skryté Markovovy modely a problém převést na problém odhadu parametrů těchto modelů.

Při modelování řeči se používají zejména tzv. levo-pravé Markovovy modely (Obr. 11), které jsou vhodné pro modelování procesů, jejichž vývoj je spojen s postupujícím časem. Princip těchto modelů je ten, že proces začíná příchodem prvního spektrálního vzoru z počátečního stavu a se vzrůstajícím časem dochází k přechodu ze stavů s nižšími indexy do stavů s vyššími indexy nebo setrvání ve stejném stavu. Proces je ukončen příchodem posledního spektrálního vzoru, kdy se proces nachází v koncovém stavu [26].

Na obrázku můžeme vidět levo-pravý Markovův model o šesti stavech, z nichž čtyři (2,3,4,5) jsou emitující stavy. To znamená, že generují pozorování a funkce rozdělení výstupní pravděpodobnosti. Zároveň tyto stavy odpovídají jednotlivým fonémům. Řekněme, že tento model je modelem určitého slova, např. *word*. Fonému *o* tedy náleží stav 3, přechodové pravděpodobnosti a_{33} a a_{34} a funkce rozdělení výstupní pravděpodobnosti b_3 [44].



Obr. 11 Levo-pravý Markovův model o 6 stavech, převzato z [84]

4.2.5 Umělé neuronové sítě

Umělé neuronové sítě (ANN) zahrnující soft computingové (SC) metody byly vytvořeny na základě jednoduchých modelů funkčních buněk nervového systému živých organismů. Stejně jako v lidském těle v tomto systému tvoří základní stavební jednotku neurony. Jednotlivé umělé neurony jsou mezi sebou spojeny, čímž vytvářejí celistvou síť schopnou paralelního zpracování informací – umělou neuronovou síť. Jednotlivá spojení pak mají různé číselné parametry (synaptické váhy, prahové hodnoty), které mohou být navrženy tak, aby předpovídali následující chování procesu ze známých dat.

Každý neuron v síti je schopen přijímat vstupní signál, zpracovat jej a vyslat výstupní signál. ANN nacházejí uplatnění v případech, kde není předem znám algoritmus řešení nebo kde je potřeba aproximovat funkční hodnoty.

Neuronové sítě mají mnoho podobností s HMM. Tam, kde Markovovy modely využívají pravděpodobnosti pro přechody stavu, používají neuronové sítě sílu a funkce spojení. Klíčovým rozdílem je, že neuronové sítě jsou v zásadě paralelní, zatímco Markovovy řetězce jsou sériové. To znamená, že obě techniky jsou velmi silné v jiném kontextu. Stejně jako v neuronové síti je úkolem nastavit vhodné váhy spojení, úkolem Markovova modelu je nalezení vhodné pravděpodobnosti přechodu a pozorování. V mnoha systémech rozpoznávání řeči jsou obě techniky realizovány společně a pracují v symbiotickém vztahu. Neuronové sítě fungují velmi dobře při učení fonémové pravděpodobnosti z vysoce paralelního audio vstupu, zatímco Markovovy modely mohou využít fonemické pozorovací pravděpodobnosti, které neuronové sítě poskytují k produkci nejpřesnější sekvence fonémů nebo slov. To je jádrem hybridního přístupu k pochopení přirozeného jazyka [45].

Akustický model lze rozdělit do dvou různých přístupů: generativní nebo diskriminativní [46]. Generativní přístup je založen na rozdělení pravděpodobnosti s danými pozorováními, zatímco diskriminativní přístup musí mít podmíněnou distribuci pomocí parametrického modelu [47]. Modely, založené na generativním přístupu jsou HMM a model směsi Gaussových distribucí (GMM) [47]. Na druhé straně ANN, hluboká neuronová síť (DNN) [48], vícevrstvé perceptrony (MLP) a metoda podpůrných vektorů (SVM) lze kategorizovat jako diskriminativní [46]. V poslední době bylo mnoho hybridních modelů navrženo pro použití v ASR [49-51]. Cílem je zlepšit výkonnost ASR kombinací sil obou přístupů [47]. Srovnání jednotlivých modelů je v Tabulka 2.

Tabulka 2 Porovnání výhod a nevýhod metod pro extrakci příznaků a akustického modelu

Metoda	Výhody	Nevýhody
MFCC	Pro výpočet se používá jednodušší DCT	Není dostatečně robustní v hlučném prostředí
PLP	Citlivější na sluchové ústrojí	Výsledné znakové vektory jsou závislé – vhodná kombinace s DNN
RASTA-PLP	Schopna vypořádat se různými druhy rušení	Není příliš robustní
GMM-HMM	Proveditelná pro nekontaminované řečové signály spolu s MFCC	Citlivost na rušení, není příliš robustní
DNN-HMM	Vyšší přesnost	Není příliš robustní pro mnoho vrstev, vyšší výpočetní nároky
SVM-HMM	Vyšší přesnost	Nevhodná pro kombinaci s metodou MFCC

5 Metody potlačování rušení

Před samotným zpracováním řečových signálů v aplikacích pro rozpoznání řeči jsou aplikovány algoritmy pro její zkvalitnění. Tyto algoritmy slouží především pro zvýšení srozumitelnosti řečového signálu a potlačení nežádoucího rušení s co možná nejmenší ztrátou užitečné informace. Algoritmy pro zkvalitnění řečového signálu lze rozdělit do dvou základních kategorií: jednokanálové a vícekanálové metody.

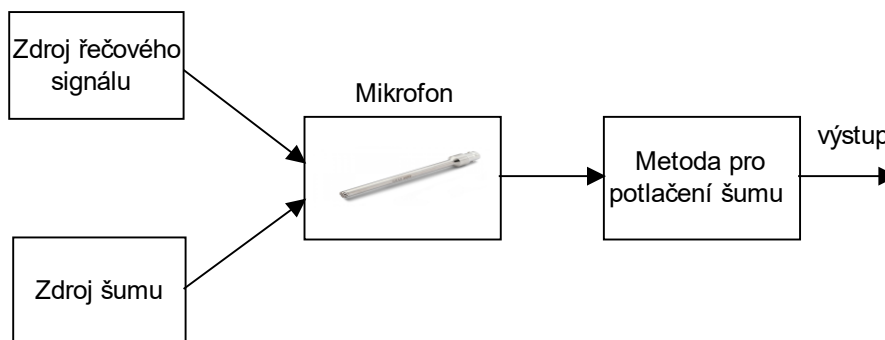
Jednokanálové metody už ze samotného názvu vyžadují pouze jeden kanál, vstup v podobě řečového signálu kontaminovaného rušením. Princip potlačení rušení je na základě rozdílných charakteristik užitečného signálu a rušení.

Druhou kategorií jsou vícekanálové metody. Snímání probíhá pomocí dvou a více mikrofonů, kde jeden je primární, který snímá zašuměný řečový signál, a zbylé jsou referenční, které snímají pouze rušení [52, 53].

5.1 Jednokanálové metody

Jednokanálové metody (Obr. 12) se uplatňují v situacích, kde druhý snímací kanál není k dispozici. Proto jsou tyto systémy v porovnání s vícekanálovými jednodušší a levnější. Metody jsou založené na předpokladu, že užitečný signál (řeč) a rušení v pozadí mají rozdílné charakteristiky. Ve svých implementacích využívají výpočtu frekvenčního spektra z dílčích segmentů signálu. Efektivita těchto metod je obvykle omezena při přítomnosti nestacionárního rušení [53, 54]. Mezi nejpoužívanější jednokanálové metody patří:

- Frekvenčně selektivní filtry typu FIR a IIR,
- metody založené na teorii Wienerovy filtrace,
- spektrální substrakce,
- vlnková transformace.



Obr. 12 Principiální schéma jednokanálového systému

5.1.1 Wienerův filtr

Podobně jako metoda spektrálního odečítání (kapitola 5.1.2) pracuje se stanovením spekter také Wienerův filtr, který klade požadavek stacionarity jak na šum, tak řečový signál. Návrh Wienerova filtru vychází ze střední kvadratické chyby (MSE) mezi originálním řečovým signálem a odhadem tohoto signálu. Odhad původního řečového signálu je výstupem Wienerova filtru [26, 55]. Spektrum hledaného signálu je stanoveno dle rovnice (5.1) [56]:

$$|\hat{X}(k)| = H(k)|Y(k)|, \quad (5.1)$$

kde $H(k)$ představuje k -tou komponentu Wienerova filtru a $Y(k)$ je spektrum zašuměného signálu. $Y(k)$ je dáno součtem spekter obou signálů, což je dáno šumem, který je aditivní složkou řeči. Filtr $H(k)$ lze pak stanovit jako:

$$H(k) = 1 - \frac{|N(k)|^2}{|Y(k)|^2} \quad (5.2)$$

5.1.2 Metoda spektrálního odečítání

Metoda spektrální subtrakce nebo též spektrálního odečítání patří mezi nejjednodušší a výpočetně nenáročné metody. Princip spočívá v odečtení spektra šumu od spektra řečového signálu kontaminovaného aditivním šumem, přičemž se předpokládá, že signál šumu není korelován s řečovým signálem. Signál je pak dále upravován v blocích, tzv. mikrosegmentech. Cole a kol. [57] k tomuto účelu používá různé šířky Hanningova okna a převádí signál pomocí FFT do frekvenčního spektra, ze kterého je následně odstraněna složka rušení a signál je inverzní FFT převeden zpět do oblasti časové.

Metoda předpokládá stacionární charakter aditivního šumu, kdy případné změny jeho spektrálních charakteristik by měly být pomalé, ve srovnání se spektrálními charakteristikami řeči [26, 58].

Nevýhodou této metody je nutnost ve sledování množství odečítané informace. Jestliže bude odečteno příliš mnoho informace, může dojít k částečné ztrátě řečového signálu. Naopak pokud bude odečteno málo informace, metoda nebude dostatečně účinná a řečový signál bude nadále obsahovat šum [58]. Dalším nedostatkem je vázání na amplitudové, resp. výkonové spektrum, kdy fázové spektrum zůstane stejné i po filtraci. Dále je efektivita metody velmi závislá na určení spektra šumu, což je v reálných podmínkách obtížné [52, 56] a pro použití ve velmi hlučném prostředí zcela nevhodná v důsledku tvorby vysokého množství hudebních tónů [59].

5.1.3 Vlnková transformace

Vlnková transformace (WT) patří do třídy MRA (Multi Resolution Analysis) jako efektivní nástroj pro analyzování nestacionárních signálů. Tato metoda používá různá rozlišení pro různé frekvence. Signál má dobré časové a špatné frekvenční rozlišení při vysokých frekvencích a tím i dobré frekvenční a špatné časové rozlišení při nízkých frekvencích. Tímto přístupem odstraňuje nedostatky diskrétní Fourierovy transformace.

Principem WT je vhodnou změnou šířky mateřské vlnky ψ dosáhnout optimálního poměru rozlišitelnosti v čase a frekvenci. Pro nízké frekvence je optimální použít vlnku širší, pro vysoké užší. Pomocí parametru měřítka s je možné měnit její šířku (dilatace) a parametrem polohy τ umístění vlnky na časové ose (translace). Mateřskou vlnku lze definovat vztahem:

$$\psi_{\tau,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right), \quad (5.3)$$

kde člen $\frac{1}{\sqrt{s}}$ slouží k normalizaci energie vlnky při změnách měřítka. Vlnka musí mít nulovou střední hodnotu, finitní energii a vhodný frekvenční obsah $\int_0^\infty \frac{|\psi(\omega)|^2}{\omega} d\omega < \infty$.

Spojité WT (CWT) je pak definována jako:

$$X(\tau, s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} x(t) \bar{\psi}\left(\frac{t-\tau}{s}\right) dt, \quad (5.4)$$

kde symbol $\bar{}$ označuje funkci komplexně sdruženou. Grafické vyjádření koeficientů (τ, s) se nazývá vlnková mapa. Vlastnostmi CWT je linearita, časová invariance a dilatace. Rekonstrukce signálu je možná také pomocí diskrétní WT (DWT), která disponuje výhodou rychlejšího výpočtu. Mateřská vlnka je pak dána vztahem:

$$\psi_{m,n}(t) = \frac{1}{\sqrt{s_0^m}} \psi\left(\frac{t-n\tau_0 s_0^m}{s_0^m}\right), \quad (5.5)$$

kde proměnné m a n určují dilataci a translaci vlnky na diskrétní mřížce. Parametry τ_0 a s_0 vyjadřují velikost kroku těchto proměnných. DWT lze pak vyjádřit jako:

$$X(m, n) = \int_{-\infty}^{\infty} x(t) \overline{\psi_{m,n}}(t) dt. \quad (5.6)$$

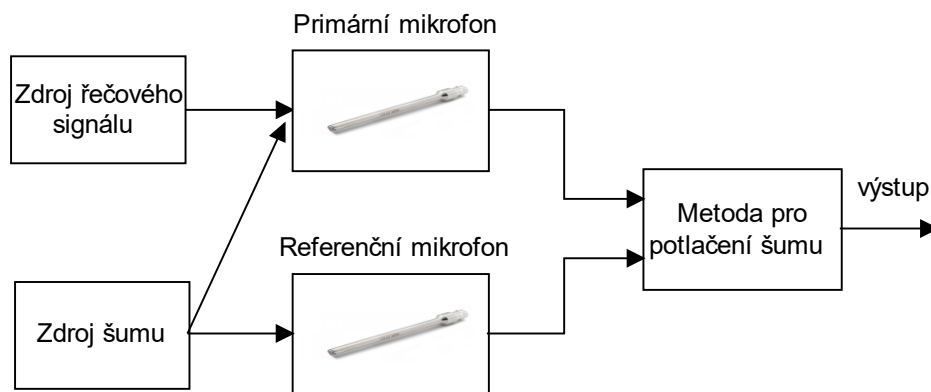
Rajeev A. a kol. [60] využili algoritmus DWT pro redukci rušení, kde byla použita metoda měkkého i tvrdého prahování. Analýza je prováděna na řečovém signálu, který je kontaminován šumem při úrovních SNR 0, 5, 10 a 15 dB. Metoda měkkého prahování dosahovala lepších výsledků na všech měřených úrovních vstupního SNR, kdy maximální zlepšení výkonu bylo o 35,16 dB. Tvrdá prahová hodnota dosahovala maximálního zlepšení o 21,71 dB.

Slavy G. a kol. [61] zkoumali využití vlnkové transformace pro redukci řečových signálů kontaminovaných rušením. Byla použita testovací databáze obsahující 720 nahrávek mužského hlasu. Soubory byly vzorkovány při frekvenci 25 kHz. Šum je přičítán k signálu řeči ve čtyřech konkrétních SNR: 0 dB, 5 dB, 10 dB, 15 dB. Díky své výpočetní složitosti se sym3 a vyšší (Symletova vlnka) ukázala být nepoužitelná pro redukci rušení v reálném čase. Naopak db3, db5 (Daubechiesova vlnka) vykazovala nejlepší vlastnosti, kdy bylo dosaženo maximálního zlepšení SNR o 14 dB.

5.2 Vícekanálové metody

Systémy využívající vícekanálové metody Obr. 13 těží z přítomnosti jednoho či více referenčních mikrofónů, které snímají pouze šum. V porovnání s jednobančovými systémy jsou tyto metody více komplexnější, avšak dosahují kvalitnějších výsledků, zejména při přítomnosti nestacionárního šumu. Do této skupiny patří [52, 53]:

- Adaptivní metody (LMS, RLS),
- principy slepé separace signálu (PCA, ICA),
- metoda tvarování paprsků.



Obr. 13 Principiální schéma dvoukanalového systému

5.2.1 Metody adaptivní filtrace

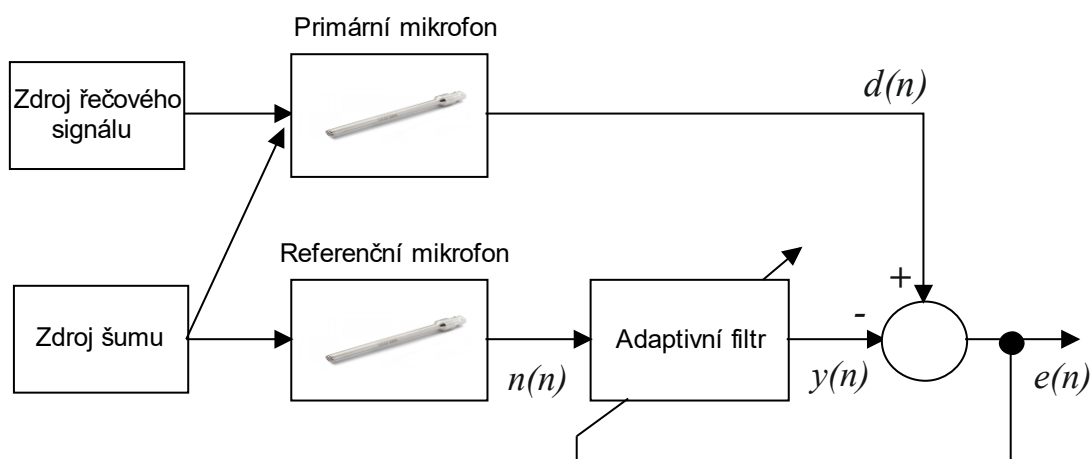
Metody adaptivní filtrace lze rozdělit do dvou základních skupin:

- Lineární filtry (LMS, RLS, ADALINE),
- nelineární filtry (Vícevrstvé neuronové sítě, Evoluční algoritmy).

Lineární filtrace je odvozena od lineárního časově invariantního systému neboli systému, ve kterém platí princip superpozice. Naopak nelineární systémy jsou takové, ve kterých tento princip neplatí.

Reálné aplikace jsou obvykle zatíženy spoustou nelinearit a neurčitostí. Ve většině případů je požadováno, aby filtr pracoval v časově proměnném a často neznámém prostředí, jehož vývoj lze těžko odhadnout. Pro odstranění těchto měnících se podmínek je nutné použít filtr, který je schopen se na tyto změny efektivně adaptovat.

Filtry, které takovou vlastnost mají, nazýváme adaptivními filtry. Základní vlastnost adaptivního filtru spočívá v kontinuální změně parametrů podle předem stanoveného algoritmu s využitím informace ze vstupu i pomocné informace, přicházející většinou z výstupu celého systému. Základním prvkem adaptivního systému je zpětná vazba, pomocí které jsou upravovány parametry samotného filtru. Cílem těchto metod je snížení účelové funkce na její minimum. Tato funkce je závislá pouze na hodnotě odchylky $e(n)$ [62, 63].



Obr. 14 Obecné blokové schéma adaptivního systému

Na Obr. 14 je uvedeno obecné schéma systému s adaptivním filtrem, kde, $y(n)$ představuje výstupní signál filtru, $d(n)$ měřený zašuměný signál, $n(n)$ šum z referenčního snímače a $e(n)$ reprezentuje odchylku, výstupního signálu od měřeného.

5.2.1.1 LMS a RLS algoritmus

Algoritmus LMS patří mezi nejrozšířenější a nejpoužívanější adaptivní algoritmy používané v současné praxi. Síla algoritmu LMS spočívá v jeho jednoduchosti a matematické nenáročnosti [64]. Tyto algoritmy jsou založeny na gradientním vyhledávacím algoritmu nazývaném také metoda největšího spádu. Závislost střední kvadratické odchylky výstupního chybového signálu adaptivního FIR filtru na koeficientech filtru je kvadratická křivka s jedním globálním minimem [65]. Výstupní rovnice je definována dle rovnice (5.7):

$$y(n) = \mathbf{w}(n)\mathbf{x}(n), \quad (5.7)$$

$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n). \quad (5.8)$$

Rekurze filtru je:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n), \quad (5.9)$$

kde μ představuje velikost kroku adaptivního filtru, $\mathbf{w}(n)$ je vektor koeficientů filtru, $\mathbf{x}(n)$ je vstupní vektor filtru.

Algoritmus RLS je základní představitel třídy rekurzivních algoritmů, které jsou založeny na teorii Kalmanovy filtrace, časovém průměrování a metodě nejmenších čtverců. Základním rozdílem oproti algoritmu LMS je vlastní statistické pojetí. Zde se pracuje s průměrnými hodnotami veličin, které jsou počítány z časových vývoji.

Struktura filtru zůstává stejná jako u LMS algoritmu, jen adaptivní proces je odlišný vzhledem k použití průměrů [66]. RLS algoritmy vynikají především vynikajícími výkony při práci v časově proměnných prostředích a extrémně rychlou konvergencí.

Nevýhodou těchto algoritmů jsou výpočetně náročné matematické operace než u algoritmů LMS. Dále RLS algoritmy mají o jeden řád vyšší složitost a jsou tedy výrazně pomalejší [67]. Výstupní rovnice RLS algoritmu je vypočítána podle následující rovnice:

$$y(n) = \mathbf{w}^T(n-1)\mathbf{x}(n). \quad (5.10)$$

Aktualizace koeficientů filtru je dána vztahem (5.11):

$$\mathbf{w}(n+1) = \mathbf{w}^T(n) + e(n)\mathbf{K}(n), \quad (5.11)$$

kde $\mathbf{w}(n)$ je vektor koeficientů filtru, $\mathbf{K}(n)$ je vektor zisků, který je definován rovnicí (5.12):

$$\mathbf{K}(n) = \frac{\mathbf{P}(n)\mathbf{u}(n)}{\lambda + \mathbf{u}^T(n)\mathbf{P}(n)\mathbf{u}(n)}, \quad (5.12)$$

kde $\mathbf{P}(n)$ je inverzní autokorelační matice. Adaptace matice se provede podle rovnice (5.13):

$$\mathbf{P}(n+1) = \lambda^{-1}\mathbf{P}(n) - \lambda^{n-1}\mathbf{K}(n)\mathbf{u}^T(n)\mathbf{P}(n). \quad (5.13)$$

5.2.2 Metoda slepé separace zdrojů

Metoda slepé separace zdrojů (BSS) v oblasti zpracování digitálního signálu spočívá v situaci, kdy je několik signálů smícháno dohromady, a úkolem je zjistit, jak vypadaly původní signály. Klasickým příkladem je „cocktail-party problém“, kdy spolu v místnosti hovoří skupina lidí a někdo se snaží poslouchat pouze jednu z probíhajících diskuzí.

Visser E. a kol. [68] zkoumali úspěšnost BSS v automobilu. Řidič vyslovoval sekvenci čísel, zatímco spolujezdec hovořil do mobilního telefonu při jízdě 40 km/h, zapnutém rádiu a ventilátoru. Pro záznam byly použity dva stereofonní mikrofony, připevněné na každé straně zpětného zrcátka (vzdálenost 15 cm) a zaznamenaná data byla vzorkována při 8 kHz. SNR směsi zaznamenané mikrofonom na straně řidiče se pohybovalo od 2 do 5 dB. Úspěšnost rozpoznání byla před použitím separace 46,9 %. Po použití BSS byla úspěšnost 72,8 %. Nejlepších výsledků dosahovala kombinace BSS a vlnkové transformace, kdy úspěšnost rozpoznání byla 79,6 %.

Visser a Lee. [69] zkoumali úspěšnost BSS v místnosti (3mx4mx6m) se dvěma směrovými mikrofony umístěné 10 cm vedle sebe. Pro generování prostorově rozloženého šumu byly použity 4 reproduktory umístěné v každém rohu místnosti. Další 2 reproduktory byly umístěny 30 cm od mikrofونů. První reproduktor vysílal sekvenci čísel, druhý sloužil jako zdroj rušení, kde vysílal předem nahrané slovo. SNR směsi zaznamenané mikrofonom byli v rozmezí -5, 0, 5 a 10 dB. Úspěšnost rozpoznání byla před použitím separace 49,34 %. Po použití BSS byla úspěšnost 84,89 %.

5.2.2.1 Metoda nezávislých komponent

Metoda nezávislých komponent (ICA) jako jedna z možností řešení „cocktail-party problému“ je statistická a výpočetní metoda pro odhalování skrytých faktorů, které jsou základem skupin náhodných proměnných, měření nebo signálů. ICA definuje model pro pozorování mnoha náhodně proměnných dat, který je typicky stanoven jako velká databáze vzorků. V tomto modelu jsou datové proměnné považovány za lineární směsi nějakých neznámých skrytých proměnných a není znám ani systém směřování. Skryté proměnné jsou považované za ne-Gaussovské a vzájemně nezávislé a jsou nazývány nezávislé komponenty pozorovaných dat. Tyto nezávislé komponenty, také nazývané zdroje nebo faktory, mohou být nalezeny metodou ICA. Analýza nezávislých komponent je povrchně spřízněná s analýzou hlavních komponent a faktorovou analýzou. Nicméně ICA je mnohem silnější metoda schopná nalézt základní faktory nebo zdroje, i když tyto ostatní metody kompletně selžou.

Máme transformaci:

$$x(k) = \mathbf{A}s(k) + \mathbf{v}(k), \quad (5.14)$$

kde \mathbf{A} představuje směšující matici. Cílem je najít matici separační, tj. matici \mathbf{H} o rozměru $N * M$, pro kterou platí $\mathbf{H} = \mathbf{H}^{-1}\mathbf{A}$. Mezi dvě základní omezení této metody patří:

- Nelze obnovit energie zdrojových signálů,
- Nelze dodržet pořadí zdrojových signálů.

Tato omezení se kompenzují tak, že se výsledná separační matice \mathbf{H} vynásobí dvěma maticemi. Matice \mathbf{P} je permutační matice, která upraví pořadí separovaných komponent a matice \mathbf{D} je diagonální matice, která upraví energie separovaných signálů. Souhrnně tedy platí:

$$\mathbf{H} = \mathbf{A}^{-1}\mathbf{D}\mathbf{P} \quad (5.15)$$

5.2.2.2 Předpoklady pro zpracování metodou ICA

Algoritmus ICA se dá rozdělit do dvou kategorií [70]. První kategorie je založena na sadě výpočtů, které minimalizují, nebo maximalizují kritériální funkci. Druhá kategorie obsahuje adaptační algoritmus, často založený na stochastické gradientní metodě, která souvisí s implementacemi adaptačního algoritmu v neuronových sítích. Statistické vlastnosti závisí na výběru účelové funkce, konvergenční vlastnosti jsou především spjaté s adaptačním algoritmem. Právě z důvodu konvergence a zjednodušení algoritmu by měly být splněny dvě základní podmínky

- Centrování vstupních dat,
- Bělení vstupních dat.

Centrování má za úkol odstranit stejnosměrnou složku z upravovaného signálu. Platí tedy:

$$x_c(k) = x(k) - \frac{1}{K} \sum_{k=1}^K x(k) \quad (5.16)$$

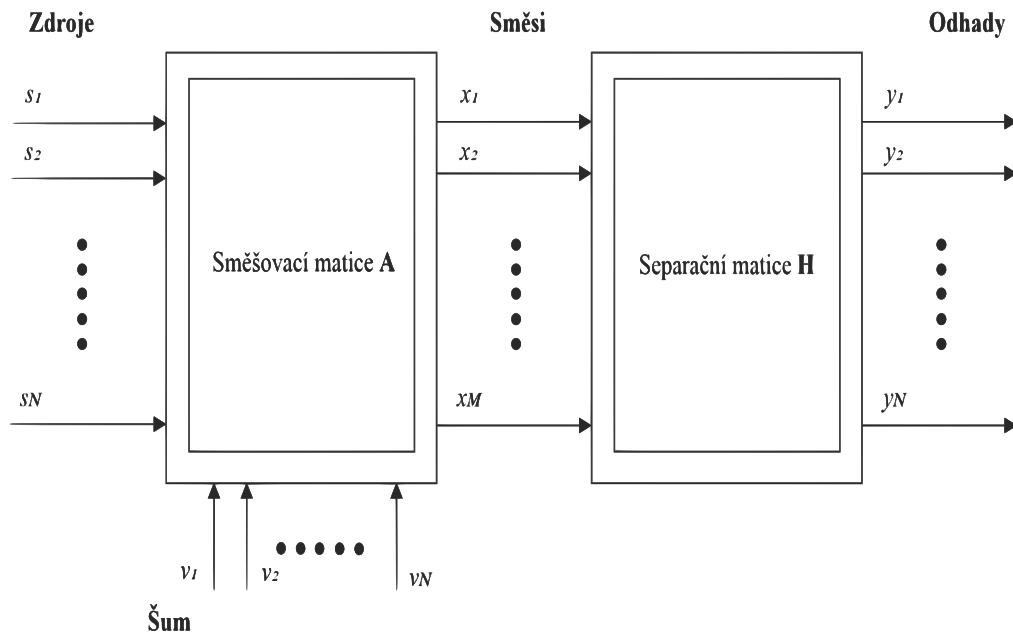
Po zpracování lze provést inverzní proces za pomoci separační matice \mathbf{H} a odhadů $y(k)$

$$y_c(k) = y(k) + \mathbf{H} \frac{1}{K} \sum_{k=1}^K x(k) \quad (5.17)$$

Bělení je takový proces před zpracování, po jehož aplikaci jsou vstupní data nekorelovaná a mají jednotkový rozptyl. Jsou-li tedy senzorové signály $x(k)$ vybíleny, pak je jejich korelační matice rovna matici jednotkové: $E\{xx^T\} = \mathbf{I}$. Tato transformace je vždy možná a může být provedena klasickou PCA (analýza hlavních komponent-Principal Component Analysis), která navíc dokáže redukovat nadbytečnou dimenzi na dimenzi odpovídající nezávislým komponentám. To je také efekt potlačení šumu. Transformace se dá napsat jako:

$$x_B(k) = \mathbf{B}x(k), \quad (5.18)$$

kde $\mathbf{B}x(k)$ označuje vybílený vektor a \mathbf{B} představuje tzv. bělicí matici rozměru $N * M$, pro kterou platí $\mathbf{B}\mathbf{B}^T = \mathbf{I}$. Pro výpočet bělicí matice lze použít metodu SVD (Singular Value Decomposition) a pomocí vlastních vektorů a vlastních čísel korelační matice vektorů směsí navrhnout bělicí matici.



Obr. 15 Základní model metody ICA

5.2.3 Analýza hlavních komponent PCA

Principal Component Analysis PCA využívá podobný model jako ICA bez vektoru aditivních šumů. Tato metoda je příbuzná rozkladu pomocí singulárních čísel a bývá označována jako Hotellingova transformace [71]. Cílem této metody je odvodit relativně malý počet de Korelovaných lineárních kombinací, které představují hlavní komponenty, které nesou co nejvíce informací původních proměnných. Hlavním účelem PCA je komprese dat a výběr příznaků, což jsou signály nesoucí nejvíce užitečné informace.

Hlavní komponenty PCA představují náhodné proměnné s maximálními odlišnostmi sestavené z lineárních směsí vstupních prvků. Tato metoda je založena na statistikách druhého řádu, oproti metodě ICA, které má náhodné proměnné s minimální entropií [72] sestrojené z lineárních kombinací vstupních prvků. PCA tedy redukuje pouze statistickou závislost druhého řádu, což představuje rozptyl. K úspěšné separaci je nutné redukovat i statistické závislosti vyšších řádů, jak umožňuje právě metoda ICA.

5.2.3.1 Rozklad pomocí singulárních čísel

V předchozí kapitole bylo řečeno, že metoda PCA je příbuzná metodě rozkladu pomocí singulárních čísel a může tak být i řešena. Podle teorie [73] [74] můžeme říct, že metoda PCA je technika výpočtu vlastních vektorů a vlastních čísel pro odhadnutou kovarianční matici:

$$R_{xx} = E\{\mathbf{x}(k)\mathbf{x}(k)^T\} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T, \quad (5.19)$$

kde $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_M]$ je ortogonální matice sloupcových vlastních vektorů a $\mathbf{\Lambda} = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_M]$ je diagonální matice obsahující M vlastních čísel.

- Váhový vektor \mathbf{v}_i maximalizuje kritérium $E = \{\sum_{i=1}^M (x_i(k))^2\} = \sum_{i=1}^M \mathbf{v}_i^T R_{xx} \mathbf{v}_i$
- Různé hlavní komponenty jsou vzájemně nekorelované
- Hlavní komponenty mají nulovou střední hodnotu $E = \{x_i(k)\} = 0$
- Rozdíl i -té hlavní komponenty je roven i -tému vlastnímu číslu kovarianční matice R_{xx}
- Hlavní komponenty jsou hierarchicky seřazeny v sestupném pořadí $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_M$

5.2.4 Metoda tvarování paprsků

Pro určení směru příchozího signálu (DOA) se používají různé metody, z nichž nejpoužívanější jsou tvarovače paprsků, tzv. beamformery (BF) [75-77]. Jsou založeny na filtraci v prostoru, kdy vybírají signály přicházející do sensorového pole z určitých směrů. Jedná se o směrovou detekci, která se provádí formováním diskretního počtu paprsků rovnající se počtu elementů v poli sensorů. Cílem metod tvarování přijímací charakteristiky je měnit váhy w_i $i = 1, 2 \dots M$ jednotlivých sensorů pro dosažení co nejlepší směrové charakteristiky v daném směru. Mezi základní tvarovače patří tzv. *delay and sum*, ze kterého jsou odvozeny další složitější tvarovače (adaptivní, GSC).

Přednost těchto metod spočívá v nezávislosti jejich parametrů na pracovních podmínkách, a především na typu vstupního signálu. Nevýhodou je malé zvýraznění vstupního signálu závislé na počtu mikrofonů [78].

Dalším aspektem je vzdálenost mikrofonů, která ovlivňuje frekvenční rozsah zpracovávaného signálu, tvar směrové charakteristiky a její frekvenční závislost.

Tabulka 3 Porovnání výhod a nevýhod uvedených metod pro odstranění rušení

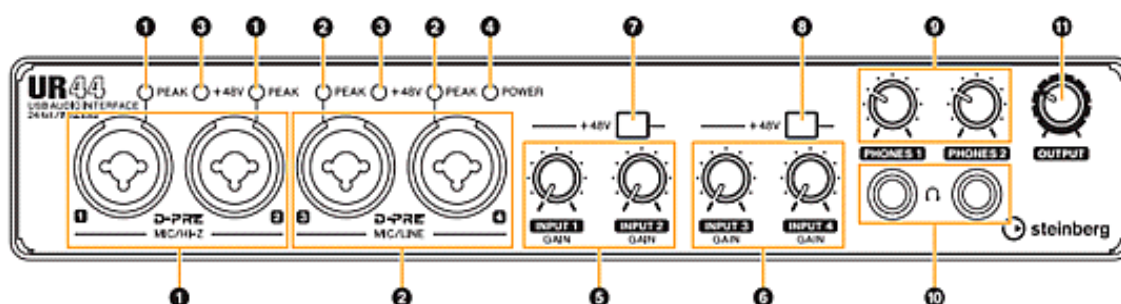
	Metoda	Výhody	Nevýhody
Adaptivní referenční zdroj	RLS	rychlost optimalizace	výpočetní náročnost
	LMS	jednoduchost	nižší rychlost
Adaptivní bez reference		pouze jeden senzor	nutnost dalšího algoritmu
Spektrální substrakce	FFT	jednoduchost	nutnost separace spektra šumu, nevhodná pro velká rušení
Vlnková transformace	DWT	Informace v časové i frekvenční oblasti, díky rozdělení komponent je snazší filtrace nestac. signálů	Větší složitost – více zdrojů, náročné na výpočet stacionárních signálů ve frekvenční oblasti
Slepá separace zdrojů	ICA/PCA	přesnost detekce šumu	výpočetní náročnost, nelze přivádět šumy Gaussovského typu
	BF	Nezávislost vstupního signálu	Špatný výkon při automatickém sledování pohyblivého zdroje zvuku

6 Popis zvukové karty Steinberg UR44 a mikrofonů RHODE NT5

V této kapitole je popsána hardwarová specifikace zvukové karty Steinberg UR44 a mikrofonů RHODE NT5. Tento hardware byl použit pro hlasové ovládání vizualizací, tvorbu databáze řečových signálů a běžně vyskytujících se rušení v oblasti SMART technologií a Průmyslu 4.0.

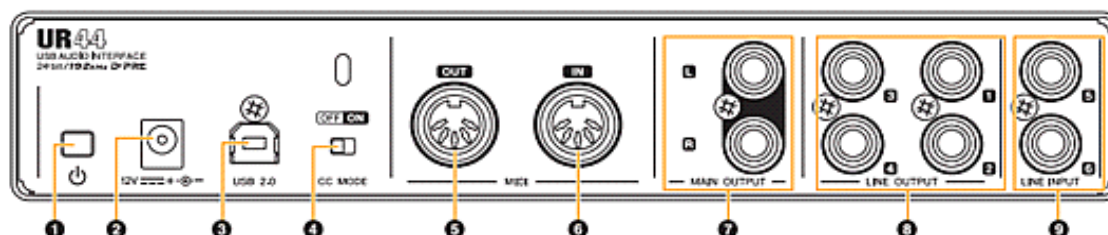
6.1 Zvuková karta Steinberg UR44

Na čelním panelu se nachází čtveřice kombinovaných XLR/JACK konektorů pro připojení mikrofonů nebo dvou elektrofonických nástrojů s vysoko impedančním (Hi-Z) výstupem (1 spodní) a dvou linkových zdrojů (2 spodní). Všechny čtyři kombinované vstupy jsou opatřeny indikací přebuzení vstupů pomocí LED (1, 2 horní). Napravo od vstupních konektorů se nachází čtveřice potenciometrů pro nastavení vstupního zisku (5, 6) a dva spínače fantomového napájení (vždy jeden společný pro dvojici vstupů), (7, 8). V pravé části čelního panelu nalezneme dva nezávislé sluchátkové výstupy (číslo 10), každý s vlastní regulací úrovně hlasitosti (číslo 9), a regulátor hlavního linkového stereofonního výstupu (11).



Obr. 16 Čelní panel zvukové karty Steinberg UR44 [79]

Zadní panel přístroje nabízí všechny ostatní vstupní a výstupní konektory. Jsou to jmenovitě hlavní linkové výstupy (7), čtveřice doplňujících linkových výstupů (8) a dva linkové vstupy (9). Všechny zmíněné vstupy a výstupy jsou symetrické (s možností připojení nesymetricky) a to v provedení JACK konektoru (TRS). Kromě audio konektorů zde najdeme také MIDI vstup (6) a výstup (5), USB 2.0 konektor typu B (3) a vstupní zdířku externího síťového napáječe s vypínačem (2). Posledním prvkem zadního panelu je spínač režimu CC (Class Compliant – pouze pro Ipad) (4).



Obr. 17 Zadní panel zvukové karty Steinberg UR44 [79]

V klasickém USB režimu jsou zapotřebí speciální ovladače pro Windows nebo Mac OS X a karta pak podporuje zažité standardy ASIO, WDM či Core Audio

Tabulka 4 Specifikace zvukové karty Steinberg UR44

Typ zvukové karty	USB
Počet analogových výstupů	6
Počet mikrofonních vstupů	4
Počet vstupů	4
Počet výstupů	4
MIDI	ANO
Fantomové napájení	+48VDC
Vzorkovací frekvence	44,1 kHz, 48 kHz, 88,2 kHz, 96 kHz, 176,4 kHz, 192 kHz
Rozlišení	až 24 bitů při maximální vzorkovací frekvenci



Obr. 18 Zvuková karta Steinberg UR44 [80]

6.2 Měřicí mikrofony RHODE NT5

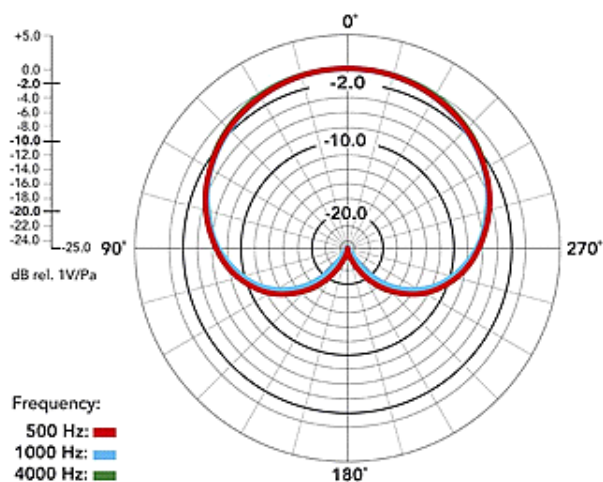
Mikrofon RHODE NT5 je malo-membránový kondenzátorový mikrofon pro záznam zvukových zdrojů, který je tvořen externě vychýleným kondenzátorem, 1/2" kapslí se zlatem pokrytou membránou, aktivní J-FET impedančním měničem s bipolárním výstupním zásobníkem a duálním napájením.

6.2.1 Fantomové napájení

Mikrofonu je potřeba dodat fantomové napájení o velikosti +24 nebo +48VDC. Fantomové napájení je stejnosměrná elektrická energie přenášená přes kabely k provozu mikrofonů, které obsahují aktivní elektronické obvody. Tento zdroj energie se nejčastěji využívá pro kondenzátorové typy mikrofonů ale také v aplikacích, kde přenos signálu probíhá přes stejné vodiče. Pro některé (ne elektretové) kondenzátory může být fantomové napájení použito pro zajištění polarizačního napětí pro samotný prvek. Toto napětí je dodáváno přes kabel směšovačem vybaveným fantomovým napájením nebo některým typem externího zdroje.

6.2.2 Kardioidní směrová charakteristika

Tato charakteristika je také známá jako ledvinová. Jedná se o nejčastěji užívanou snímací charakteristiku. Mikrofon snímá nejvíce ve vlastní ose. Ostatní směry jsou potlačovány a ze zadní strany je snímání minimalizováno.



Obr. 19 Směrová charakteristika mikrofonu RHODE NT5 [81]

Tabulka 5 Specifikace mikrofonu RHODE NT5

Akustický princip	Gradient tlaku
Úroveň akustického tlaku	143 dB
Aktivní elektronika	J-FET impedančním měničem s bipolárním výstupním zásobníkem
Směrová charakteristika	Kardioidní (ledvinová)
Frekvenční rozsah	20 Hz – 20 KHz
Výstupní impedance	100Ω
Možnosti napájení	24VDC nebo 48VDC
Citlivost	-38 dB re 1 Volt/Pascal (12 mV @ 94 dB SPL) +/- 2
Ekvivalentní úroveň šumu	16dBA
Výstup	XLR
Váha	101 g



Obr. 20 Mikrofon RHODE NT5 [81]

7 Experimentální část

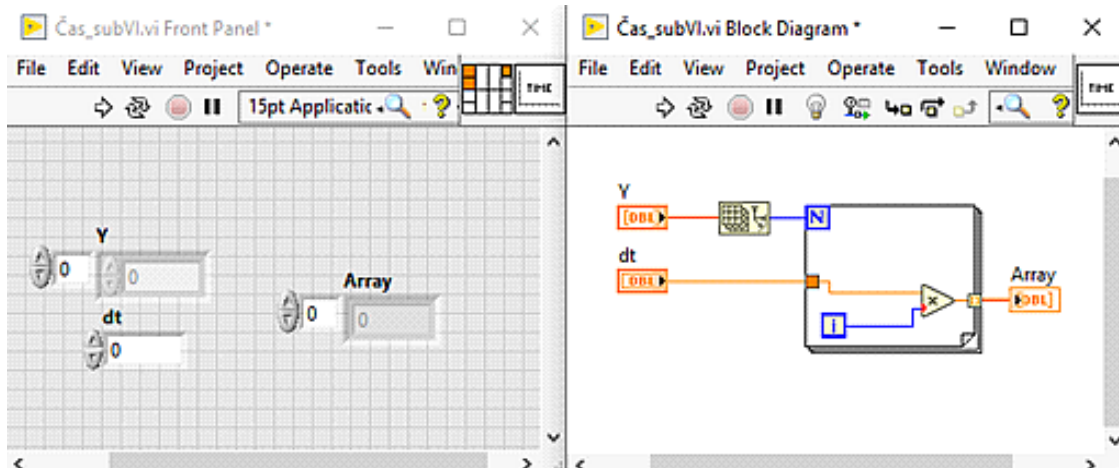
Experimentální část sestává ze tří bloků, kdy je nejprve uveden stručný popis vybraného softwaru, volby vhodné architektury, popis přenosu dat, komunikace se zvukovou kartou a výběr rozpoznávače. Ve druhé části je popsána realizace databáze rušení a funkční popis všech aplikací. V poslední části jsou vybrány metody pro potlačení rušení pro následné vyhodnocení úspěšnosti rozpoznávače.

7.1 Software LABVIEW

Grafické vývojové prostředí LabVIEW (Laboratory Virtual Instruments Engineering Workbench) je produktem americké firmy National Instruments, který umožňuje programování ve specifickém grafickém programovacím jazyku “G”. Tímto se stává intuitivní i pro nezkušené, začínající programátory umožňující programovat i bez hlubší znalosti syntaxe. Prostředí je tedy na úrovni např. jazyka C, ale na rozdíl od něj není orientován textově ale graficky. Výsledný produkt tohoto vývojového prostředí se nazývá virtuálním instrumentem (Virtual Instrument, zkráceně VI), protože svým charakterem a činností připomíná klasický přístroj ve své fyzické podobě. Virtuální instrument je tedy základní jednotkou každé aplikace vytvořené v tomto vývojovém prostředí a obsahuje [82]:

- Interaktivní grafické rozhraní (Graphical User Interface – GUI) - tzv. čelní panel, který simuluje čelní panel fyzického přístroje. Obsahuje objekty, jako jsou ovládací a indikační prvky, pomocí kterých lze řídit běh aplikace, zadávat parametry a získávat informace o zpracovaných výsledcích.
- Blokové schéma, ve kterém se definuje posloupnost vyhodnocení jednotlivých složek programu – samotný algoritmus programu, jejich propojení a parametry. Každá komponenta obsahuje vstupní a výstupní připojovací body. Jednotlivé připojovací body lze propojit s prvky na panelu pomocí propojovacího nástroje tzv. wiring tool.
- Podřízené virtuální instrumenty, tzv. subVI. Instrument má modulární, hierarchickou strukturu. To znamená, že jej lze používat samostatně, jako celý program, nebo jeho jednotlivé subVI. Součástí každého VI je jeho ikona, kterou je reprezentován v blokovém diagramu a konektor s připojenými místy pro vstupní a výstupní parametry.

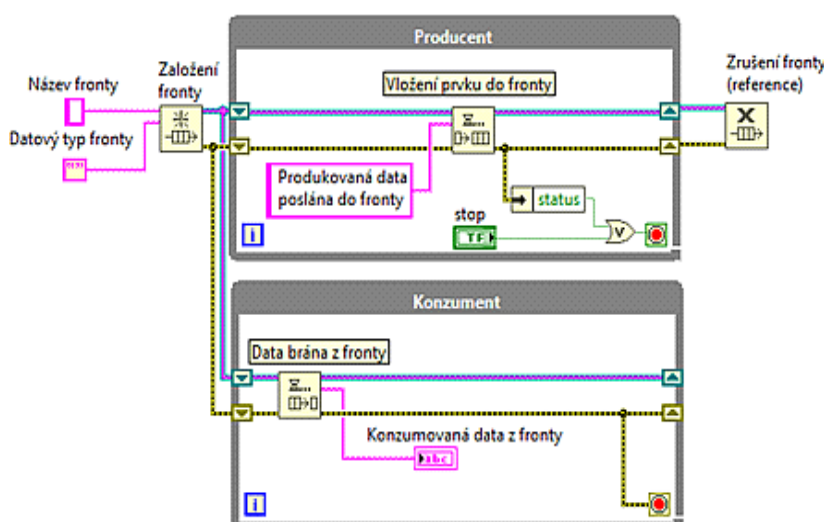
Sekvence běhu programu je dána datovým tokem. Uzel blokového diagramu se provede, když obdrží všechny požadované vstupy. Po vykonání uzlu, vytvoří výstupní data a předá data dalšímu uzlu v cestě datového toku. Pohyb dat přes uzly určuje pořadí provedení VI a funkcí na blokovém diagramu.



Obr. 21 Pohled na uživatelské rozhraní vývojového prostředí LabVIEW

7.1.1 Architektura aplikací

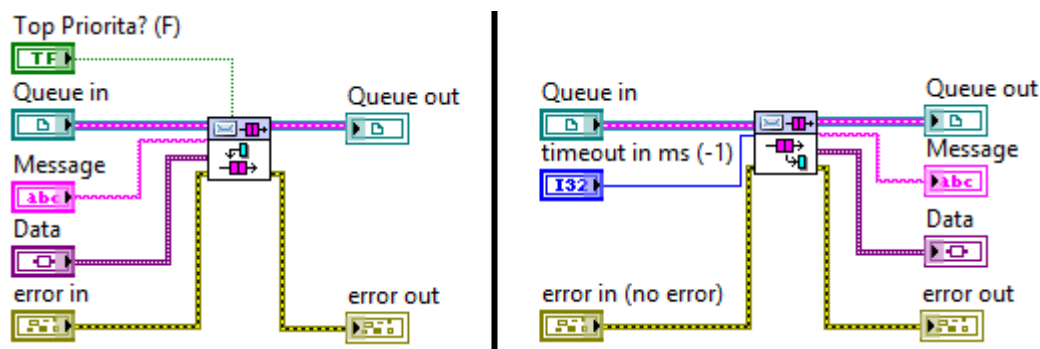
Požadavkem v zadání diplomové práce bylo, aby program běžel co nejvíce modulárně, byl co nejjednodušší v budoucnu rozšiřitelný a zároveň uživatelsky intuitivní. Není žádáno, aby implementace nových částí software trvala dlouho a byla složitá. Architektura pro všechny aplikace je tedy navržena dle doporučeného návrhového vzoru QMH (Queued Message Handler), což je architektura software na principu producent/konzument. Hlavní funkční vlastností této architektury je, že jednotlivé pod-části programu jsou provedeny pouze na základě podnětu, kterým může být například reakce na změnu hodnoty ovládacího prvku čelního panelu. Program pak nepracuje s vyčtenou hodnotou periodicky, ale vykonává danou část kódu jenom v případě, že hodnota byla změněna operátorem nebo programově (např. kontinuální volání jedné smyčky pro výčet dat).



Obr. 22 Ukázka návrhového vzoru producent/konzument

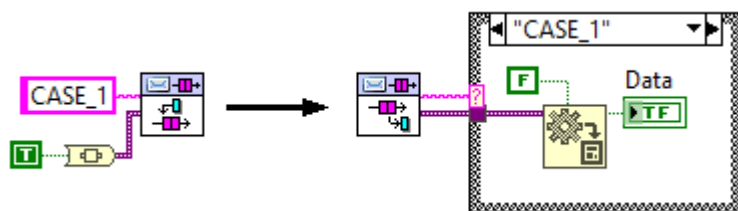
7.1.1.1 Komunikace mezi procesy

Jak již bylo řečeno, byla zvolena architektura producent/konzument, kde producent v reakci na události generuje zprávy. Konzument je vyčítá a podle jejich obsahu spouští různé části kódu. Komunikaci a předávání dat mezi smyčkami zajišťují fronty, které jsou zásobníkové paměti typu FIFO (First in, first out), tedy první zařazená zpráva bude z fronty vyčtena jako první. Jako datový typ pro komunikaci mezi smyčkami byl zvolen klastr, který obsahuje dva prvky. Prvním prvkem je *Message* datového typu *String*. Tento prvek je určen pouze k režii daného stavu. Druhým prvkem jsou *Data* datového typu *Variant*, což je univerzální datový typ umožňující přenášet všechny datové typy. Dále je možné nastavit prioritu zpráv. Pokud bude priorita *True*, bude zpráva vložena na začátek fronty a vykonána jako první. Pokud bude *False*, bude zpráva vložena na konec fronty. U funkce pro vyčítání dat je umístěn *Timeout* pro čekání na zprávu. Pokud je nastaven na -1 bude čekat nekonečně dlouhou dobu. Každé kladné číslo je čekání v milisekundách.



Obr. 23 Vlevo data enqueue, vpravo data dequeue

Jakýkoliv datový typ se přetypuje na *Variant* a následně se zašle spolu se zprávou. Na straně „příjemce“ této zprávy se zase *Variant* přetypuje zpátky na daný datový typ. Tuto konverzi je nutné si pohlídat a správně naprogramovat, protože pokud uživatel nedodrží daný datový typ, tak dojde k vyvolání výjimky o nedodržení datového typu – vygenerování chybové hlášky a zastavení programu.

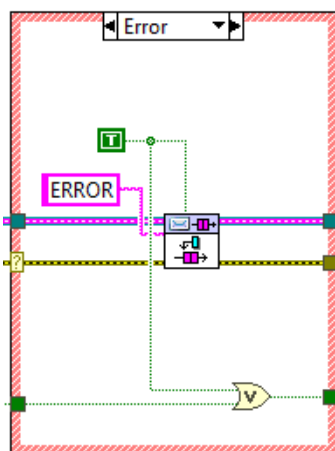


Obr. 24 Vlevo jsou data zapsána, vpravo vyčtena

7.1.2 Zachytávání chyb a jejich řešení

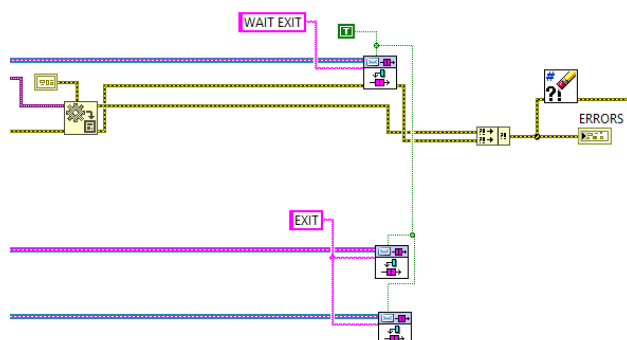
Zachytávání chyb (Handling Errors) je řešeno ve všech aplikacích pomocí tzv. *error wire*. Není žádoucí, aby při vzniku chyby v jedné z několika smyček běžel program pouze na určitý výkon, ale aby celá aplikace byla zastavena a řádně ukončena. To je docíleno pomocí volání stavu *ERROR*.

Pokud je zachycena chyba, je vyhodnocen chybový stav *ERROR* a poslána prioritní zpráva „*ERROR*“, viz Obr. 25.



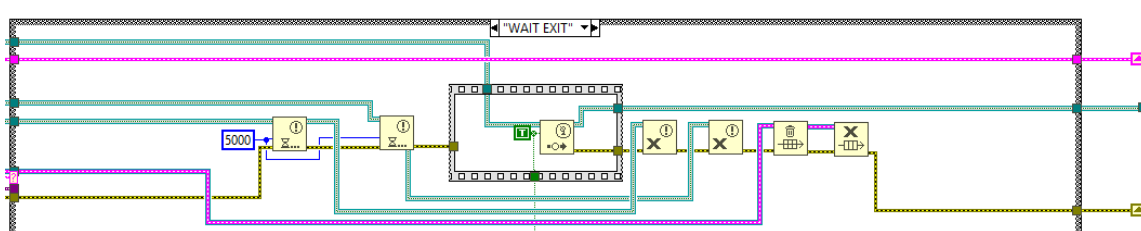
Obr. 25 Vyhodnocení a zaslání zprávy „*ERROR*“

Následně dojde v hlavní smyčce k vykonání stavu *ERROR*. Ten zašle prioritní zprávy ostatním smyčkám o ukončení, odstraní danou chybu a přejde do stavu *WAIT EXIT*.



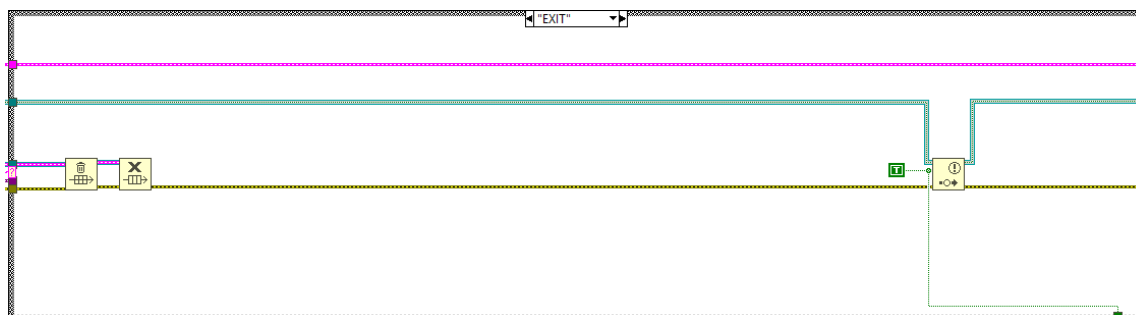
Obr. 26 Hlavní smyčka zpracovávající stav *ERROR*

Ve stavu *WAIT EXIT* čeká 5 sekund na zprávu z ostatních smyček o tom, že byly řádně ukončeny pomocí notifikací. Pokud se řádně ukončily, dojde k vygenerování uživatelské události tzv. *User event*, která ukončí uživatelské rozhraní. Následně jsou uvolněny reference na notifikace, vyprázdněna a uvolněna fronta, a nakonec vypnuta smyčka.



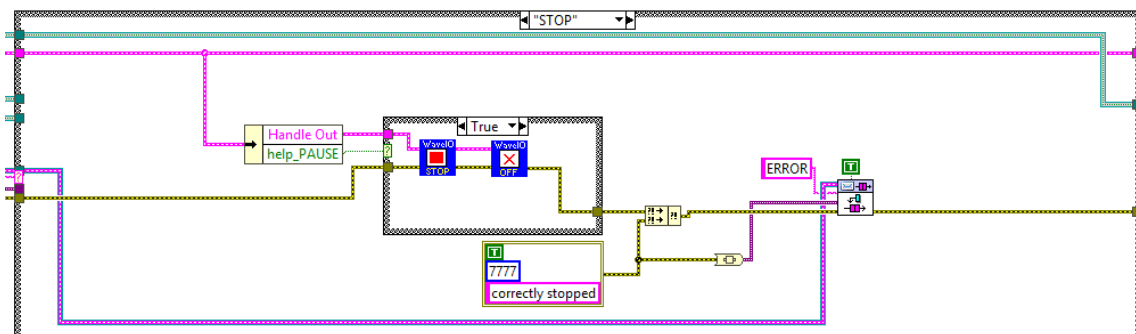
Obr. 27 Stav *WAIT EXIT* v hlavní smyčce

Ve stavu *EXIT* dojde k vyprázdnění příslušné fronty, její uvolnění a zaslání notifikace o ukončení smyčky.



Obr. 28 Generování notifikace jedné z vedlejších smyček

Ukončení aplikace uživatelem probíhá stejným způsobem tak, že je úmyslně vygenerována chyba s číslem 7777 a hlášením „*correctly stopped*“.



Obr. 29 Korektní ukončení aplikace

7.1.3 Komunikace mezi LabVIEW a zvukovou kartou

Aby bylo možné komunikovat v LabVIEW se zvukovou kartou Steinberg UR44, byla použita volně dostupná knihovna vytvořená Christianem Zeitnitzem [83]. Spojení mezi LabVIEW a zvolenou zvukovou kartou se provádí pomocí standardního Windows API. To platí pro zvukové karty PCI (Peripheral Component Interconnect) i zvukové karty s připojením USB, kde je podporován paralelní provoz více zvukových karet. Další VI umožňují ovládat Windows Mixer za účelem výběru zdroje a cíle zvukové karty, ztlumení nebo vypnutí zvuku a sledování nebo ovládání úrovně hlasitosti.

7.1.3.1 Instalace

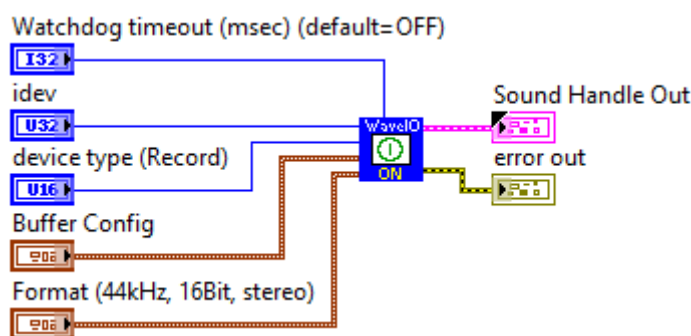
Pro použití knihovny *WaveIO* stačí zkopírovat soubory *waveio.dll* a *waveio.llb* do příslušného projektu nebo do adresáře LabVIEW *user.lib*. Při následujícím spuštění vývojového prostředí se VI objeví v paletě funkcí pod *User library*. Veškeré aplikace jsou napsány ve verzi LabVIEW 64bit, je tedy zapotřebí nahrát správné knihovny jinak nebude kód spustitelný.

Pro komunikace se zvukovou kartou je nutné implementovat následující funkce:

- **WaveIO_Open:** Založí referenci zvukové karty pro nahrávání nebo přehrávání. Zařízení musí být spuštěno před vlastním čtením nebo zápisem dat. Funkce vrátí chybu, pokud zařízení neexistuje, je již otevřeno nebo vybráno.

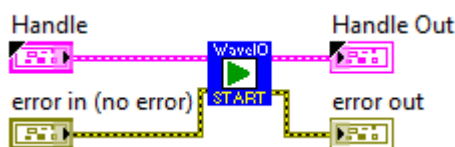
Watchdog zavře zařízení, pokud není v daném časovém okně zjištěno žádné volání do sdílené knihovny (DLL). Pro každé zařízení se spustí samostatné sledovací vlákno.

Pro výběr pouze podmnožiny fyzického kanálu může být poskytnut seznam povolených čísel kanálů (počínaje od 1). Počet kanálů se v tomto případě automaticky vypočítá z počtu položek v seznamu. Pokud je první položka seznamu nulová, je položka pro počet kanálů interpretována jako bitová mapa povolených kanálů (maska 32bitových kanálů).



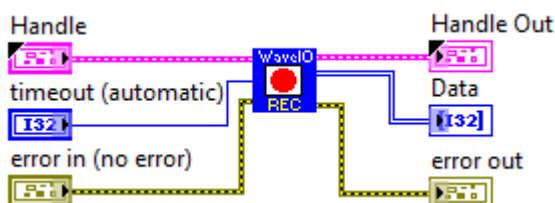
Obr. 30 Funkce WaveIO_Open

- **WaveIO_Start:** Spustí úlohu měření na zvukové kartě



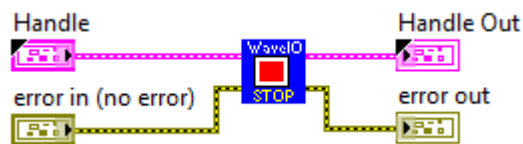
Obr. 31 Funkce WaveIO_Start

- **WaveIO_Record:** čeká a načítá data ze zvukové karty.



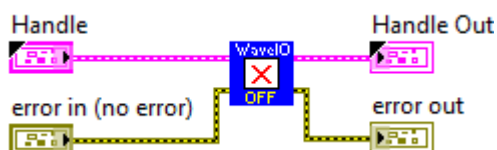
Obr. 32 Funkce WaveIO_Record

- **WaveIO_Stop:** Zastaví úlohu měření na zvukové kartě. Tato funkce je požadována před zavřením reference



Obr. 33 Funkce WaveIO_Stop

- **WaveIO_Close:** Ukončení reference měření na zvukové kartě

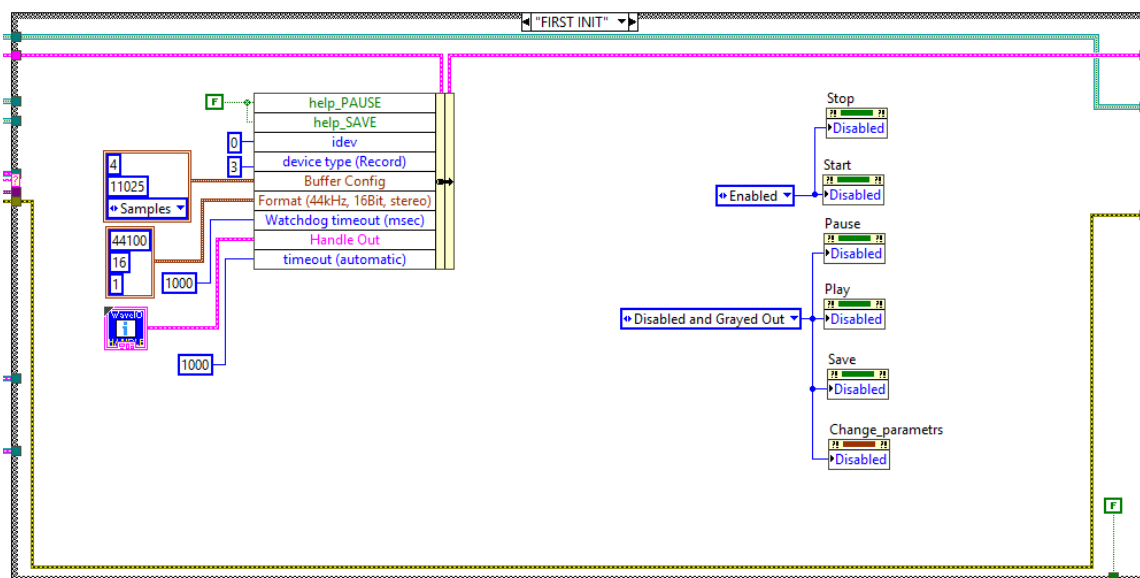


Obr. 34 WaveIO_Close

7.2 Tvorba databáze nahrávek rušení

Pro záznam nahrávek rušení byla vytvořena aplikace, která sestává z grafu pro zobrazení časového okna nahrávek, z grafu zobrazující předešlé hodnoty (historie nastavena na maximální hodnotu, tedy 65535 vzorků) a pěti tlačítek pro obsluhu programu.

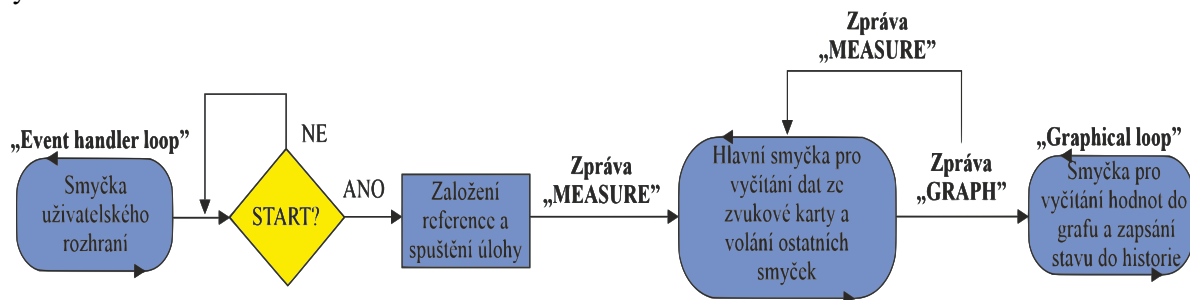
Po spuštění programu je provedena inicializace front, notifikací, uživatelských událostí a je poslána zpráva „FIRST INIT“ do všech smyček pro inicializaci jejich interních proměnných. V této fázi aplikace čeká na zmáčknutí tlačítka *START* a spuštění měření.



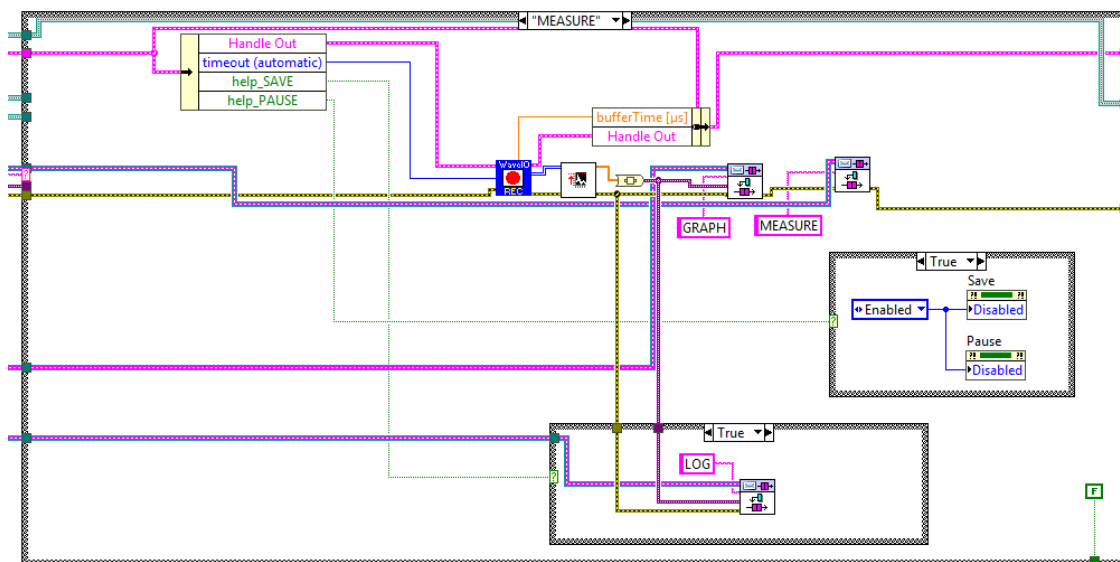
Obr. 35 Inicializace hlavní smyčky

7.2.1 Spuštění měření

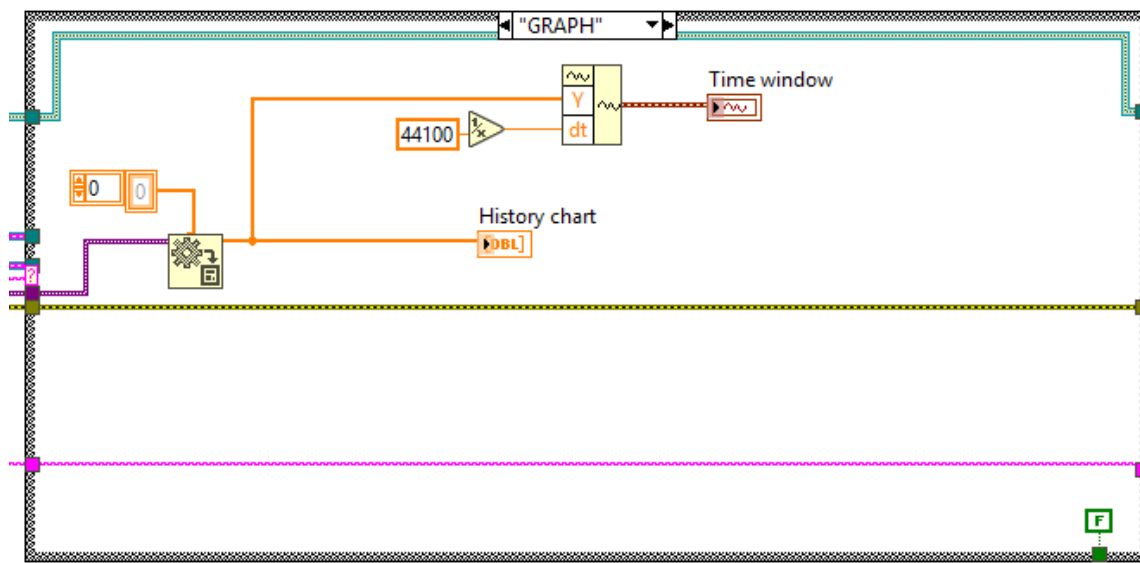
Po zmáčknutí tlačítka *START* dojde k založení reference měření, jeho spuštění a je poslána zpráva „*MEASURE*“ pro vyčítání dat ze zvukové karty. Data jsou posílána do smyčky („Graphical loop“), která obsluhuje zobrazení do grafů. Na závěr je opět poslána zpráva „*MEASURE*“ pro kontinuální vyčítání dat.



Obr. 36 Diagram algoritmu po zmáčknutí tlačítka *START*



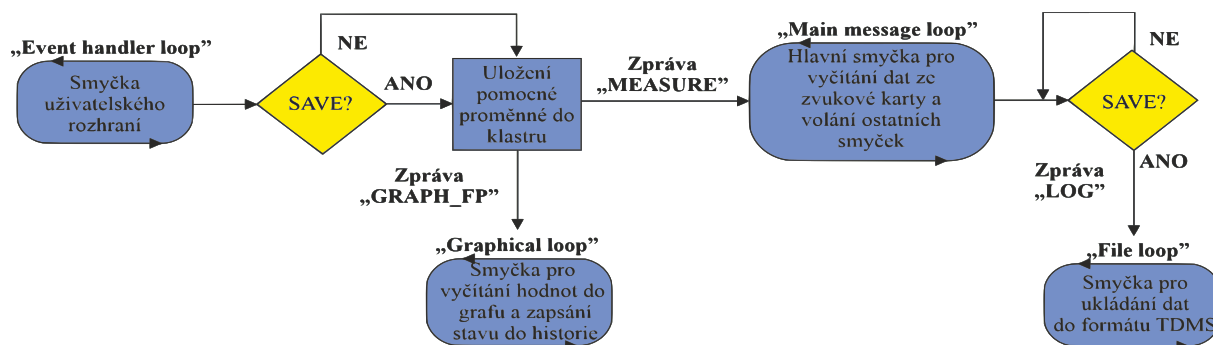
Obr. 37 Stav *MEASURE* v hlavní smyčce



Obr. 38 Vyčítání hodnot do grafů v grafické smyčce

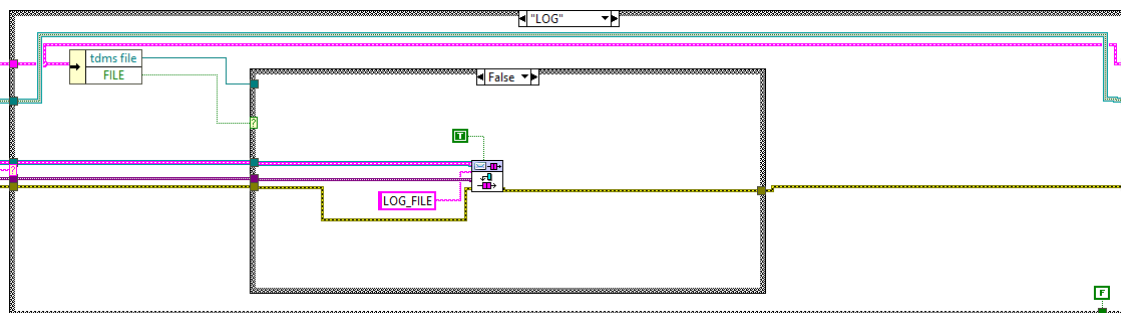
7.2.2 Ukládání dat

Ukládání dat probíhá pomocí tlačítka *SAVE*, který funguje jako přepínač. Pokud dojde k jeho sepnutí, je zapsána hodnota *True* do pomocné proměnné a přenesena do stavu „*MEASURE*“ pomocí klastru. V tomto stavu je vyhodnoceno, zdali je pomocná proměnná ve stavu *True* nebo *False*. Pokud je ve stavu *True* je volána smyčka pro ukládání dat pomocí zprávy „*LOG*“. Pokud dojde k přepnutí do stavu vypnuto, je zapsána hodnota *False* do pomocné proměnné a ve stavu *MEASURE* je volání smyčky pro ukládání ukončeno.



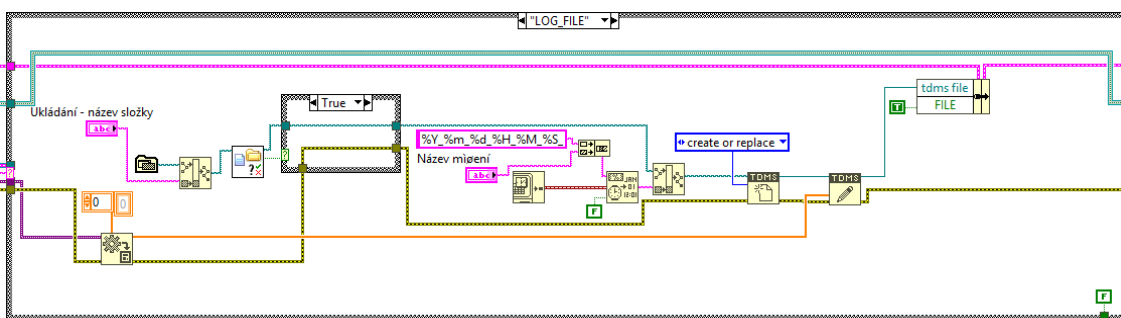
Obr. 39 Diagram algoritmu pro ukládání dat

Ve stavu „*LOG*“ je pomocí proměnné *FILE* vyhodnoceno, jestli již byla založena složka pro ukládání nebo ne. Pokud ano, jsou data zapsána do souboru. Pokud ne, je volán stav „*LOG_FILE*“.

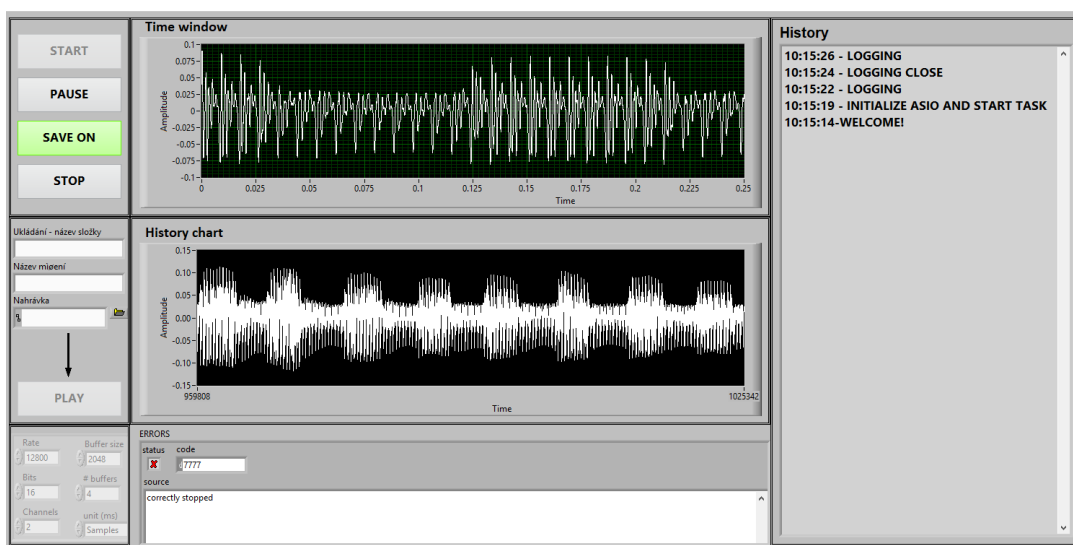


Obr. 40 Vykonání stavu LOG

V tomto stavu dojde k založení složky do kořenového adresáře VI, dále je soubor pojmenován podle aktuálního datumu a času, kdy uživatel může přidat za toto razítko vlastní pojmenování. Následně jsou data zapsána do formátu TDMS a do pomocné proměnné *FILE* je uložena hodnota *True*. Dále smyčka obsahuje *Timeout* nastavený na 1 sekundu, který na základě přijatých dat vyhodnocuje, jestli již bylo ukládání ukončeno nebo ne. Pokud do 1 sekundy neobdrží žádná data, vykoná stav *WAIT*, který uvolní referenci ukládání a do pomocné proměnné *FILE*, zapíše *False*.



Obr. 41 Vykonání stavu LOG_FILE

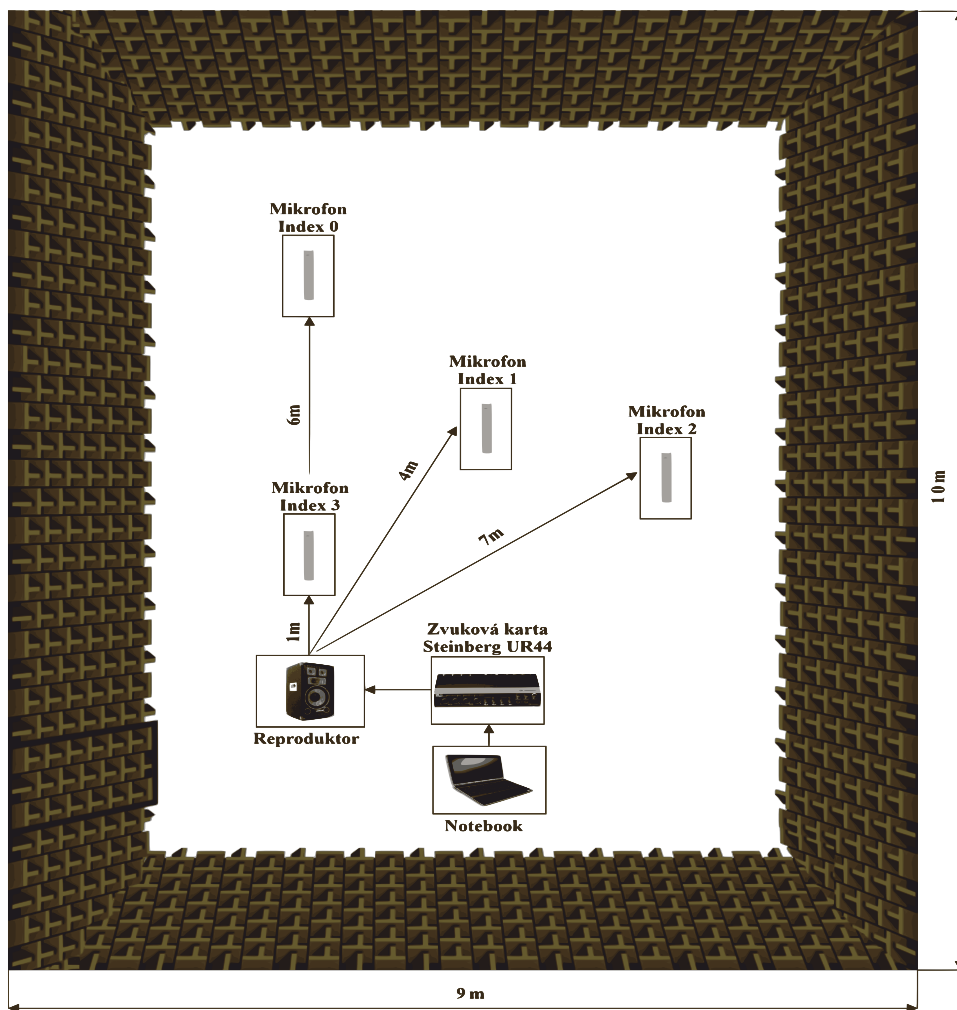


Obr. 42 Čelní panel aplikace pro tvorbu nahrávek rušení

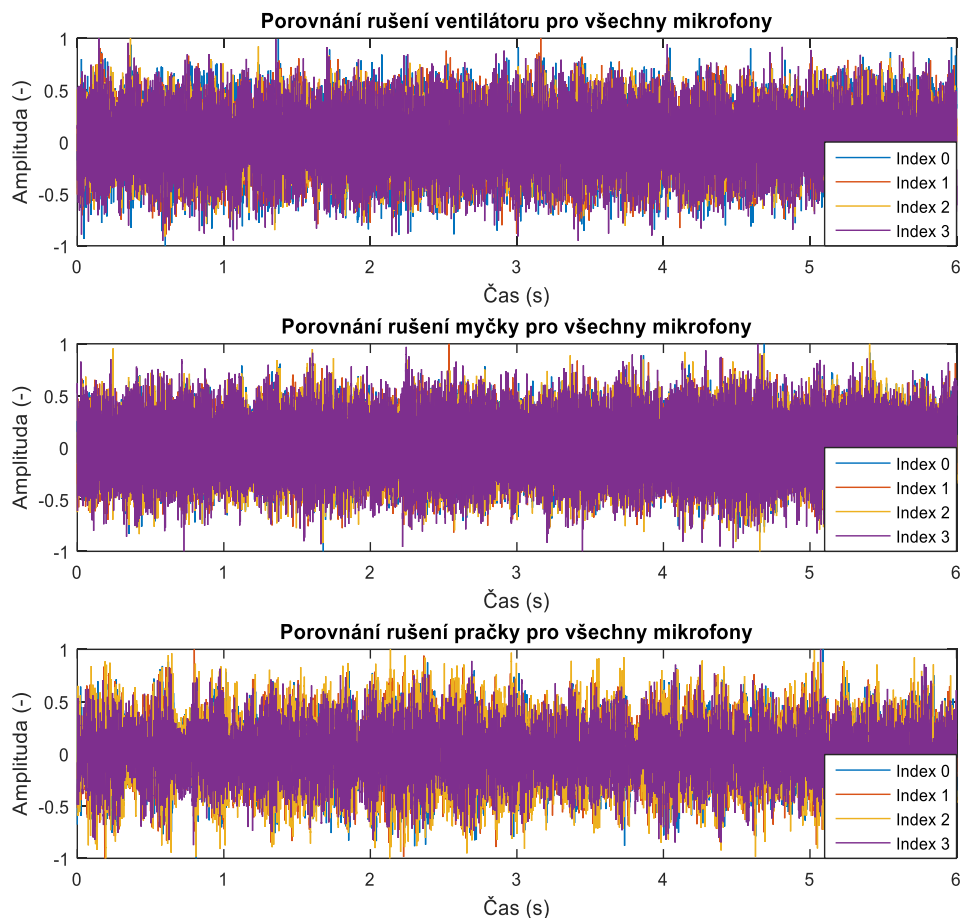
7.2.3 Měření v polobezodrazové komoře

Pro vytvoření nahrávek řeči a některých rušení (pračka, myčka, vysavač, ventilátor) byla použita polobezodrazová komora (rozměry 9x10m) nacházející se v kampusu VŠB – Technické univerzity Ostrava, která slouží pro posuzování účinků nežádoucího hluku. Jedná se o univerzální akustické pracoviště určené k měření hluku velmi tichých dějů proudění kapalin a plynů nebo hlukových dějů spojených s mechanickou činností. Komoru lze využívat také například pro optimalizaci chráničů sluchu nebo při testování zařízení na rozpoznávání hlasu. Bezodrazivost je zajištěna obkladem stropu a stěn hlubokými absorpčními klíny z akusticky pohltivého materiálu, v němž se postupující akustická energie mění v teplo, čímž je znemožněn její odraz a ovlivnění probíhajícího měření.

Pro měření referenčních nahrávek řeči byl nejprve umístěn mikrofón do zadní části komory (index 0). Pro měření rušení byl využit reproduktor, který byl postupně pouštěn na výkon 20, 50 a 100 %, a sada čtyř mikrofónů viz obr. 22. Na indexu 0 se nachází primární mikrofón ve vzdálenosti 7 metrů od reproduktoru. Index 1 slouží pro záznam rušení ve vzdálenosti 4 metrů, index 2 pro záznam ve vzdálenosti 7 metrů a index 3 pro záznam ve vzdálenosti 1 metru.



Obr. 43 Umístění mikrofónů, reproduktoru a zvukové karty v polobezodrazové komoře po měření rušení

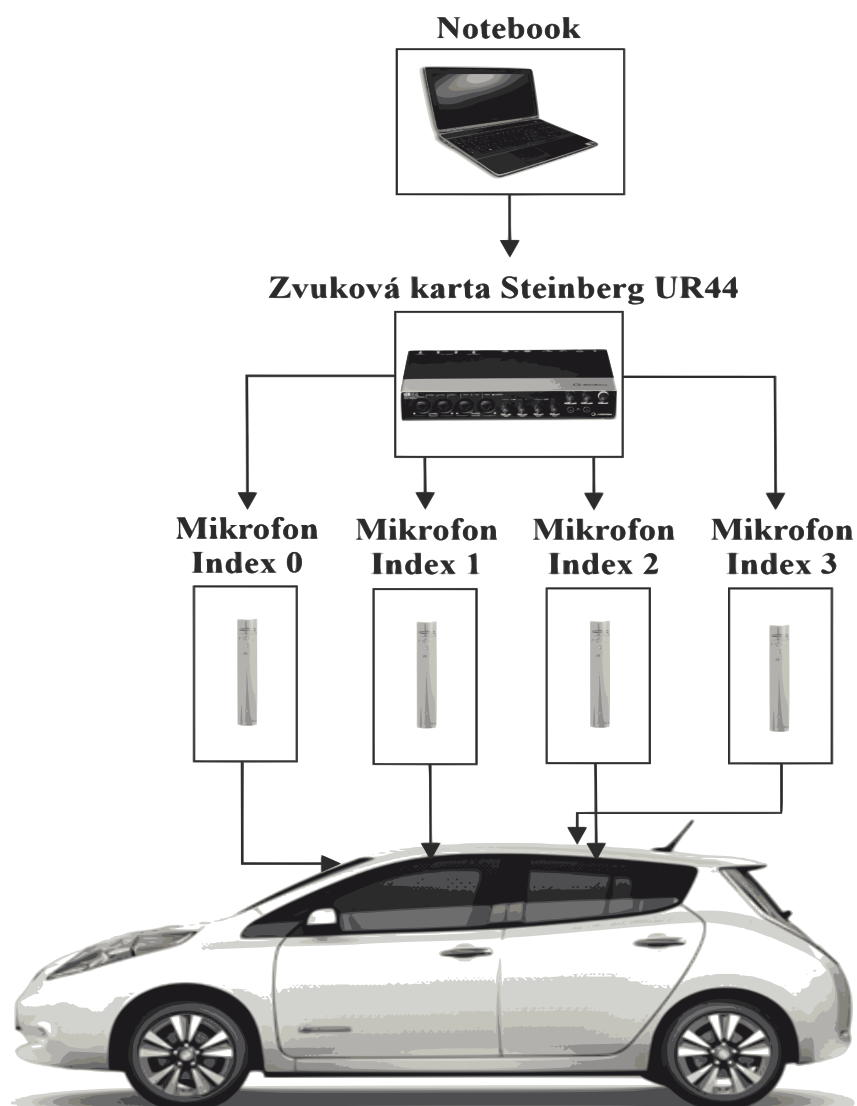


Obr. 44 Ukázka naměřených rušení v polobezodrazové komoře

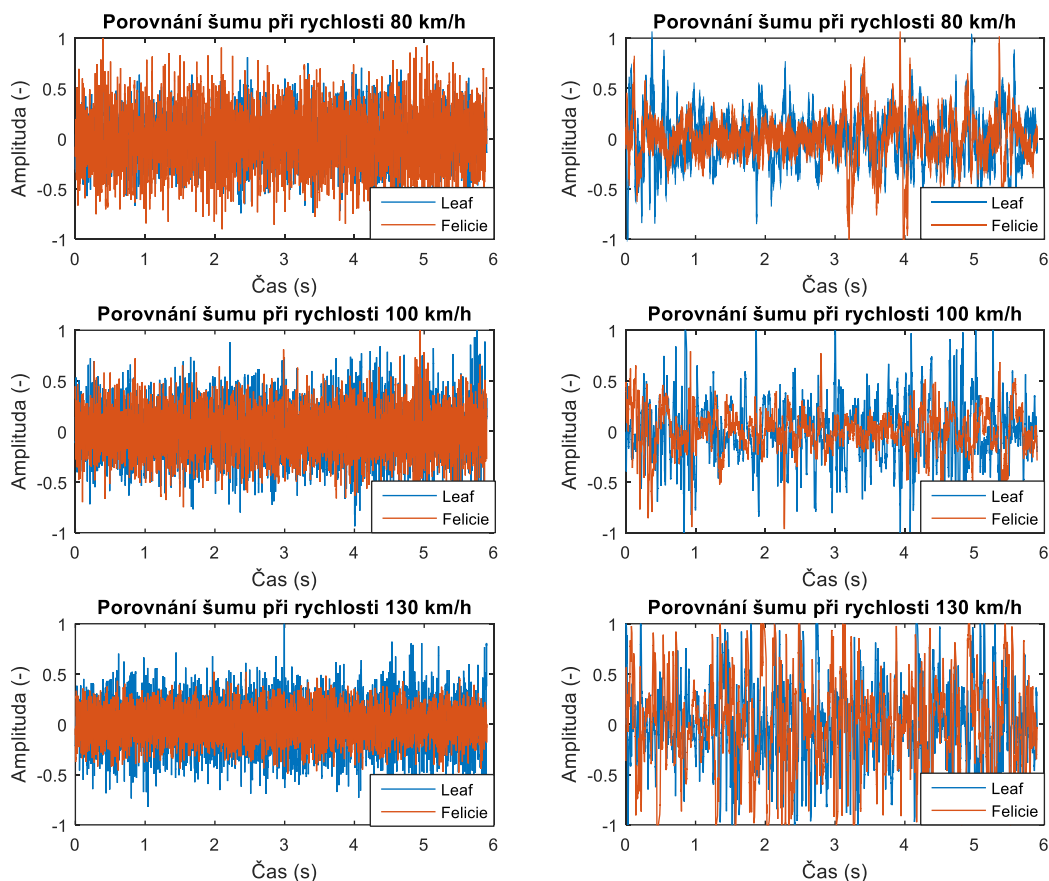
7.2.4 Měření v automobilech

Pro realizaci nahrávek byla využita dvě vozidla. Prvním z nich byl automobil značky Nissan Leaf. Jedná se o elektromobil s celkovým výkonem 80kW, maximální rychlostí 144 km/h a maximálním dojezdem až 250 km. Druhým vozem byla Škoda Felicie s celkovým výkonem 50kW a maximální rychlostí 152 km/h.

Pro záznam rušení byl primární mikrofon (index 0) umístěn nad zpětným zrcátkem v kabině vozu a nasměrován do prostoru řidiče. Zbylé tři referenční mikrofony byly připevněny v prostoru oken, kdy mikrofon s indexem 1 byl umístěn k oknu na straně spolujezdce, mikrofon s indexem 2 byl umístěn k levého zadního oknu (okno za řidičem) a poslední mikrofon snímal rušení u pravého zadního okna. Nahrávky byly pořízeny v rychlostech 20, 50, 80, 100, 130 km/h nejprve pro zavřená okna, následně pro otevřená a v neposlední řadě pro kombinace jednotlivých oken (vždy jedno otevřené, zbytek zavřené). Zapojení zvukové karty a umístění jednotlivých mikrofonů je znázorněno na Obr. 45.



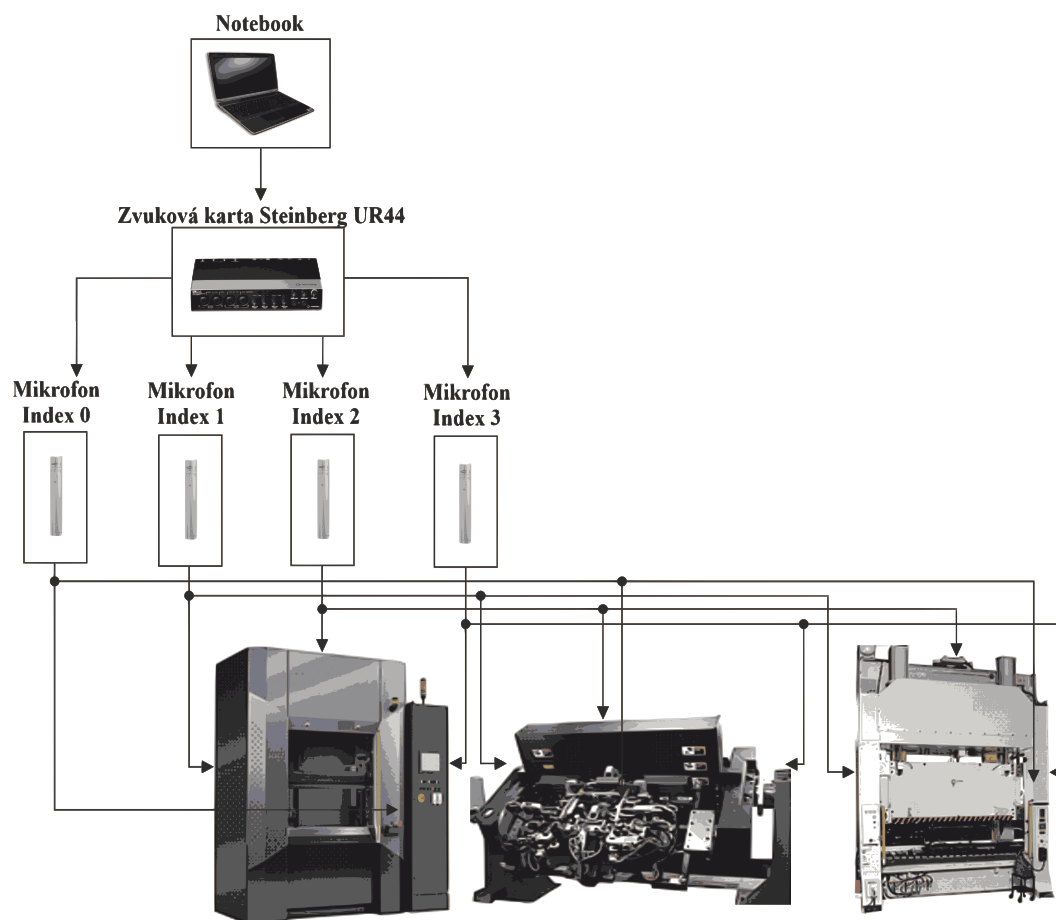
Obr. 45 Realizace měření ve automobilech



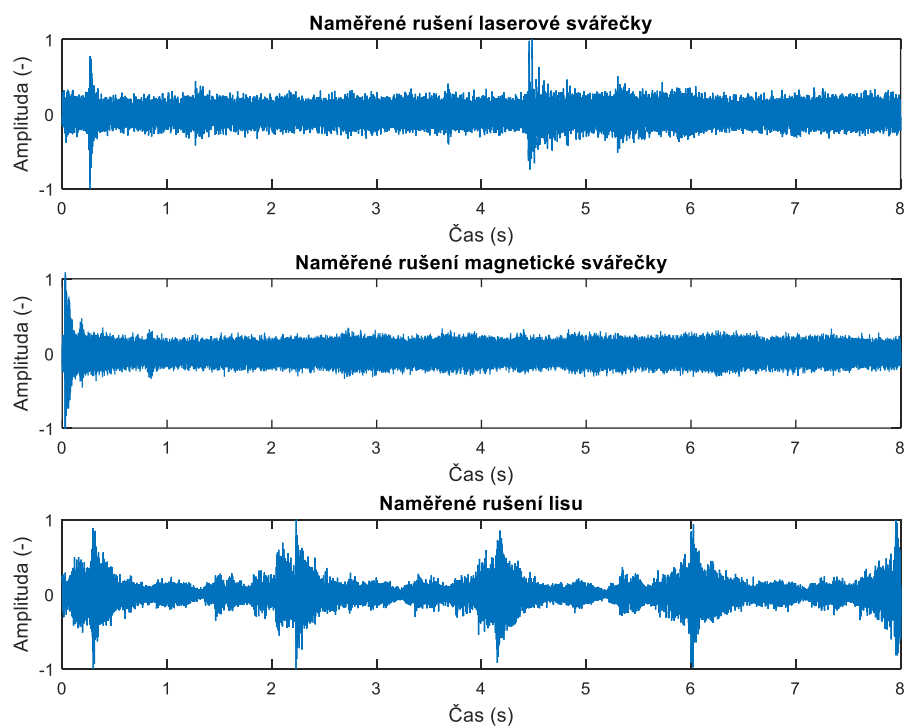
Obr. 46 Ukázka průběhu rušení. Primární mikrofon (index 0) zavřená okna vlevo, otevřená okna vpravo

7.2.5 Měření ve výrobní lince

Měření bylo realizováno ve firmě BROSE sídlící v Kopřivnici, jejichž hlavní produkcí jsou sedadlové struktury, elektrické motory, pohony, zámky zadních a bočních dveří. Měření byly zdroje rušení laserové svářečky, magnetické svářečky a lisu. Primární mikrofon s indexem 0 byl umístěn v prostoru, kde operátor obsluhoval daná zařízení. Referenční mikrofony s indexem 2 a 3 byly umístěny po stranách zařízení a referenční mikrofon s indexem 2 byl umístěn v zadní části zařízení, viz Obr. 47.



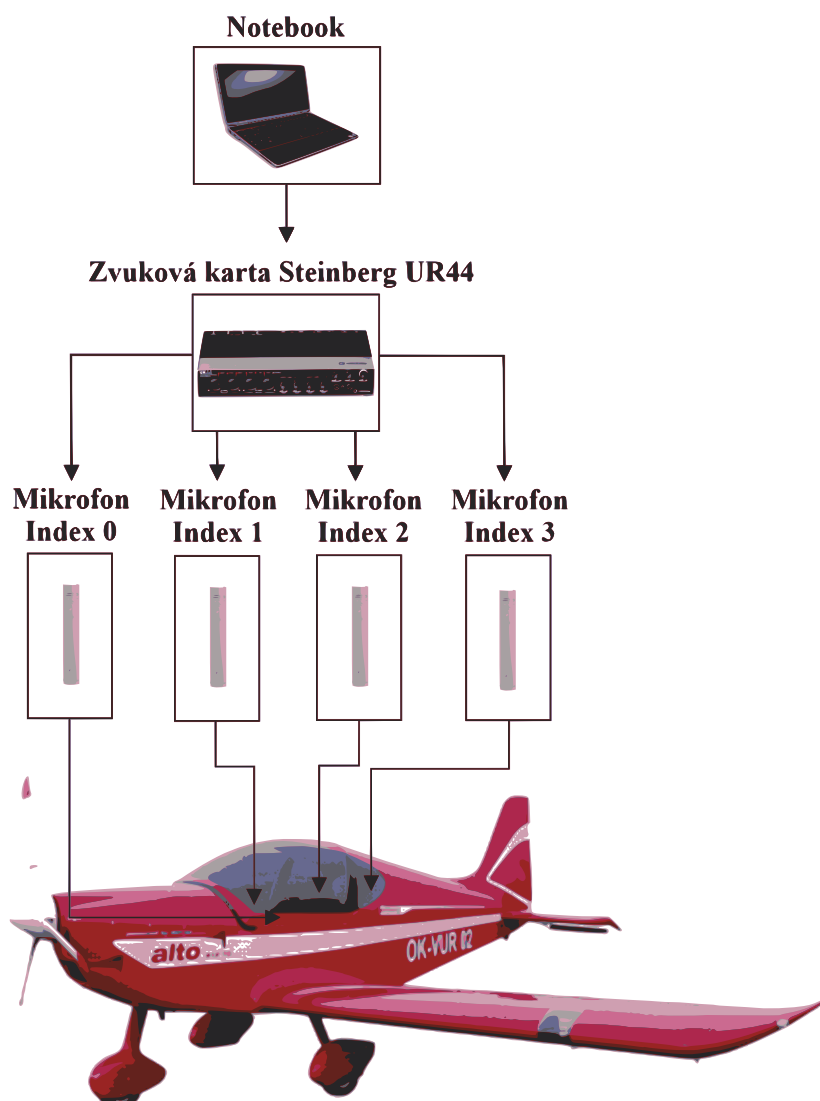
Obr. 47 Realizace měření ve výrobní lince



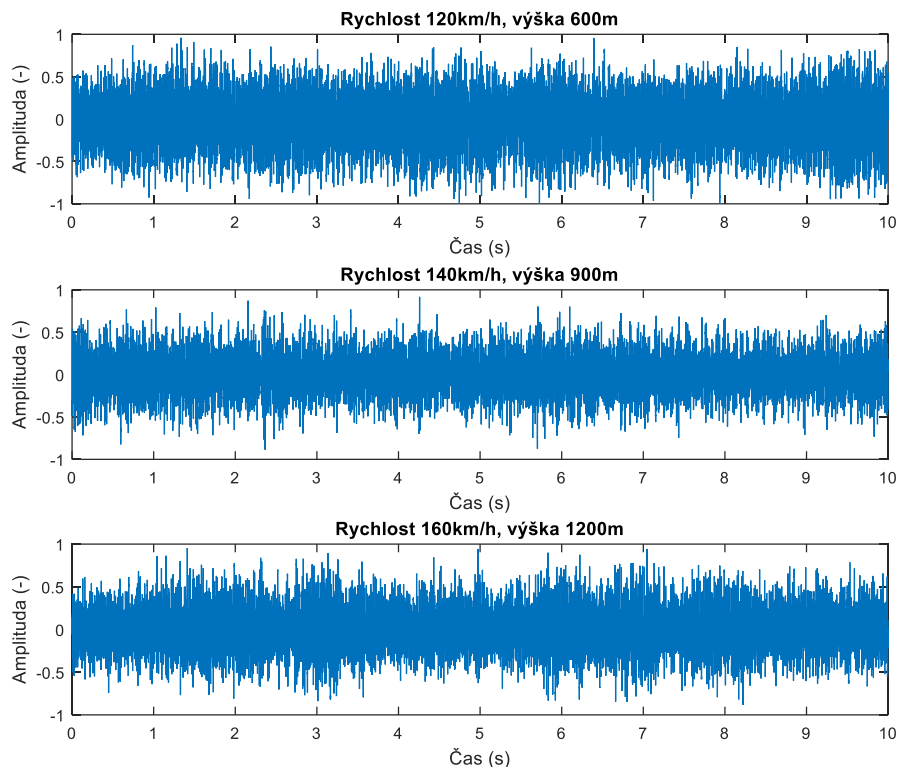
Obr. 48 Ukázka naměřených rušení laserové/magnetické svářečky a lisu

7.2.6 Měření v letadle

Posledním měřením rušení bylo v letadle Alto 912GT. Jedná se o dvoumístné letadlo s motorem Rotax 912, cestovní rychlostí 100 km/h a maximální rychlostí 250 km/h. Nahrávky byly pořízeny v rychlostech 120, 140 a 160 km/h ve výšce 600, 900 a 1200 metrů. Mikrofon s indexem 0 byl umístěn ve dveřích u pilota, mikrofon s indexem 1 ve dveřích pasažéra, mikrofon s indexem 2 byl u úst pilota a poslední mikrofon v úložném prostoru za zády pilota.



Obr. 49 Realizace měření v letadle Alto 912GT



Obr. 50 Ukázka naměřených rušení při různých rychlostech a výškách

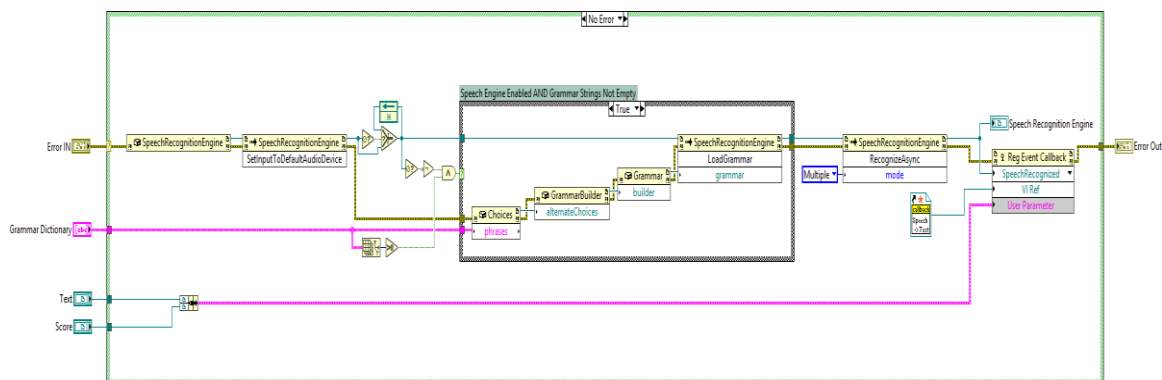
7.3 Vizualizace

Jedním z požadavků diplomové práce bylo vytvořit modely, které budou korespondovat s naměřenými nahrávkami z reálného prostředí. Na základě tohoto požadavku byli vytvořeny tři vizualizace, jimiž jsou interiér automobilu, inteligentní domácnost a výrobní linka. Pro hlasové ovládání vizualizací byl použit komerčně dostupný rozpoznávač ve Windows od firmy Microsoft.

7.3.1 Převod řeči na text

Aby bylo možné komunikovat s rozpoznávačem je zapotřebí nainstalovat ovladač *Speech SDK 5.1*, který funguje na principu převodu hlasového příkazu na text. Jedná se o volně dostupný software, který umožňuje vývojářům aplikovat syntézu a rozpoznávání řeči ve Windows z různých programovacích jazyků. Nevýhodou je omezená slovní banka pro češtinu, kdy jsou podporovány jen určité jazyky (angličtina, čínština, francouzština, němčina atd.). Z toho důvodu je nutné mít uživatelské prostředí ve Windows nastaveno na anglický jazyk.

Pro komunikaci mezi LabVIEW a rozpoznávačem bylo použito volně dostupné VI (Speech recognition engine), kterému stačí pouze definovat vstupní pole příkazů ve formátu *string* do vstupního konektoru *Grammar dictionary*, podle kterého jsou porovnávány hlasové příkazy. Poté je na výstupu „*Recognized command*“ výsledek porovnaného příkazu převedeného do textové podoby.

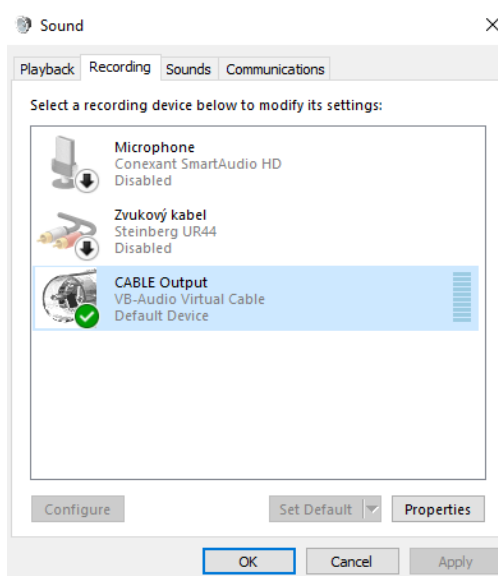


Obr. 51 VI pro komunikaci s rozpoznávačem

7.3.2 Propojení pomocí virtuálního kabelu

Proto, aby hlasové příkazy, ke kterým bylo přičteno rušení bylo možné filtrovat a zasílat tak odfiltrovaná data do rozpoznávače, bylo nutné nainstalovat program VB-CABLE (Virtual Audio Cable). Jedná se software pro přenos audio streamu mezi aplikacemi nebo zařízeními. Vytváří sadu virtuálních zvukových zařízení (tzv. Virtual Cables), kde každý z nich je složen z dvojice vstupních/výstupních koncových bodů. Libovolná aplikace pak může poslat audio stream na výstupní část virtuálního kabelu a další aplikace může přijmout tento stream prostřednictvím jeho vstupní části. Všechny přenosy jsou prováděny digitálně a nedochází ke ztrátě kvality. Program je vhodný pro nahrávání audio výstupu aplikací v reálném čase, či přenos zvukového streamu do jiné aplikace, která jej dále zpracovává.

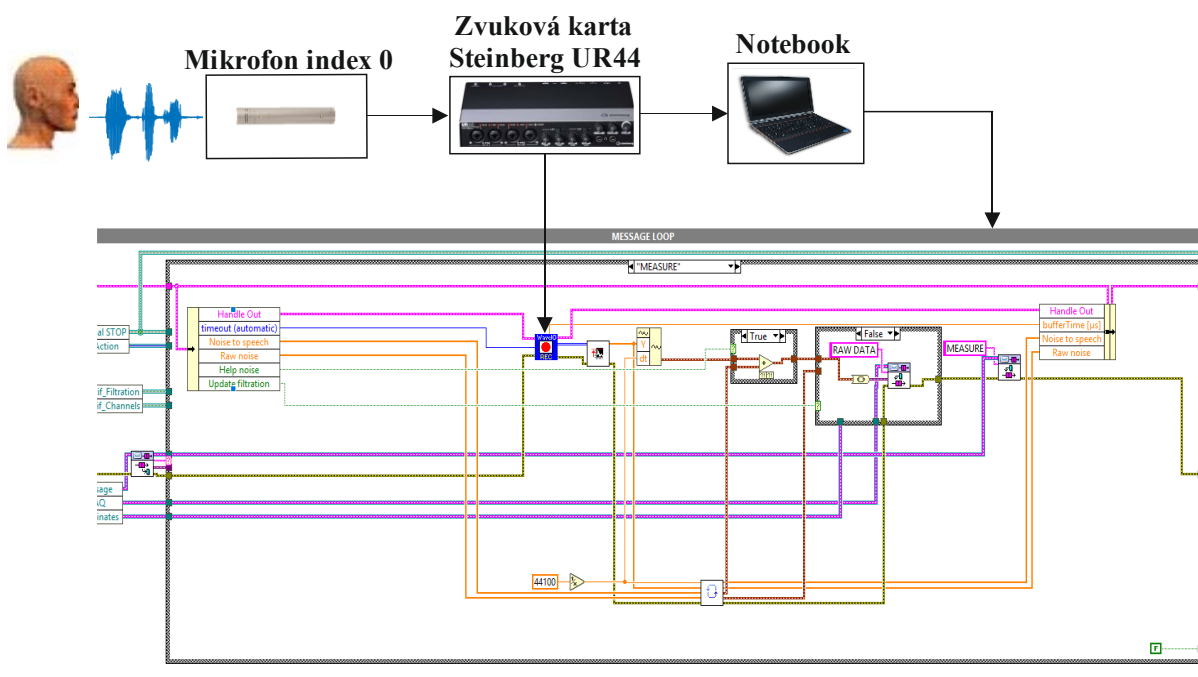
V počítači je potřeba v nastavení zvuku povolit nahrávání z tohoto virtuální propojení a ostatní zařízení zakázat (mikrofon integrovaný v notebooku a zvukovou kartu Steinberg UR44, popřípadě jiná zařízení).



Obr. 52 Nastavení virtuálního kanálu do výchozího stavu

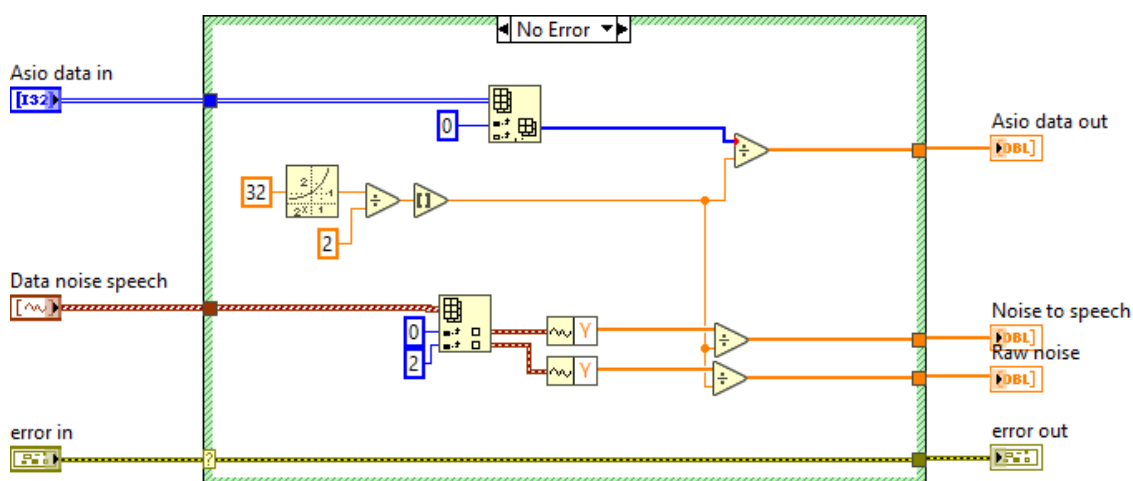
7.3.3 Hlavní smyčka pro vyčítání dat

Pro hlasové ovládání vizualizací byl použit jeden mikrofon, který je nastaven na index 0. Veškeré vizualizace obsahují hlavní smyčku, jejichž úkolem je kontinuální vyčítání dat, jejich škálování, přičítání rušení k řeči a zasílání těchto dat do dalších smyček. Měřicí řetězec je znázorněn na Obr. 53.



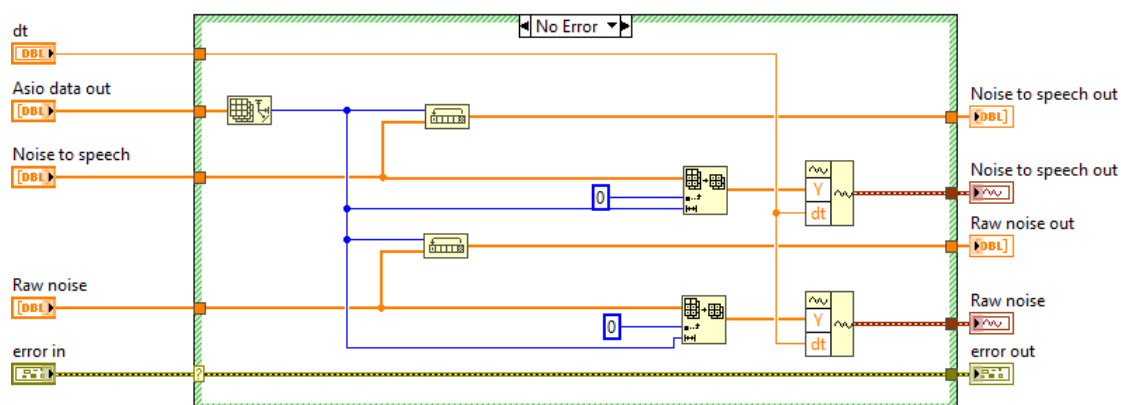
Obr. 53 Hlavní smyčka pro kontinuální vyčítání dat

Jak již bylo popsáno více, jsou výstupní data ze zvukové karty reprezentována znaménkovým int32 rozlišením ($-2^{31} + 2^{31} - 1$). Proto je nutné zpětně přepočítat hodnoty amplitudy tak, že jsou nejprve indexovány vstupní signály a poděleny hodnotu $2^{32}/2$, viz Obr. 54.



Obr. 54 Škálování hodnot ze zvukové karty

Pokud je aktivován stav pro načtení nahrávek rušení, jsou tato rušení přičtena z databáze k reálné řeči. Jelikož nahrávky rušení mají přibližně 5sekundovou délku, ale sběr dat ze zvukové karty může probíhat libovolnou dobu, bylo potřeba synchronizovat délky těchto signálů. Toho je docíleno pomocí subVI *Rotace sumu*, kdy je nejprve zjištěno časové okno sběru dat, pomocí kterého je definováno časové okno nahrávek rušení. To je následně vyčteno a posunuto o další časové okno vpřed. Jakmile dojde k posunutí na konec nahrávky, začíná celý cyklus od začátku.



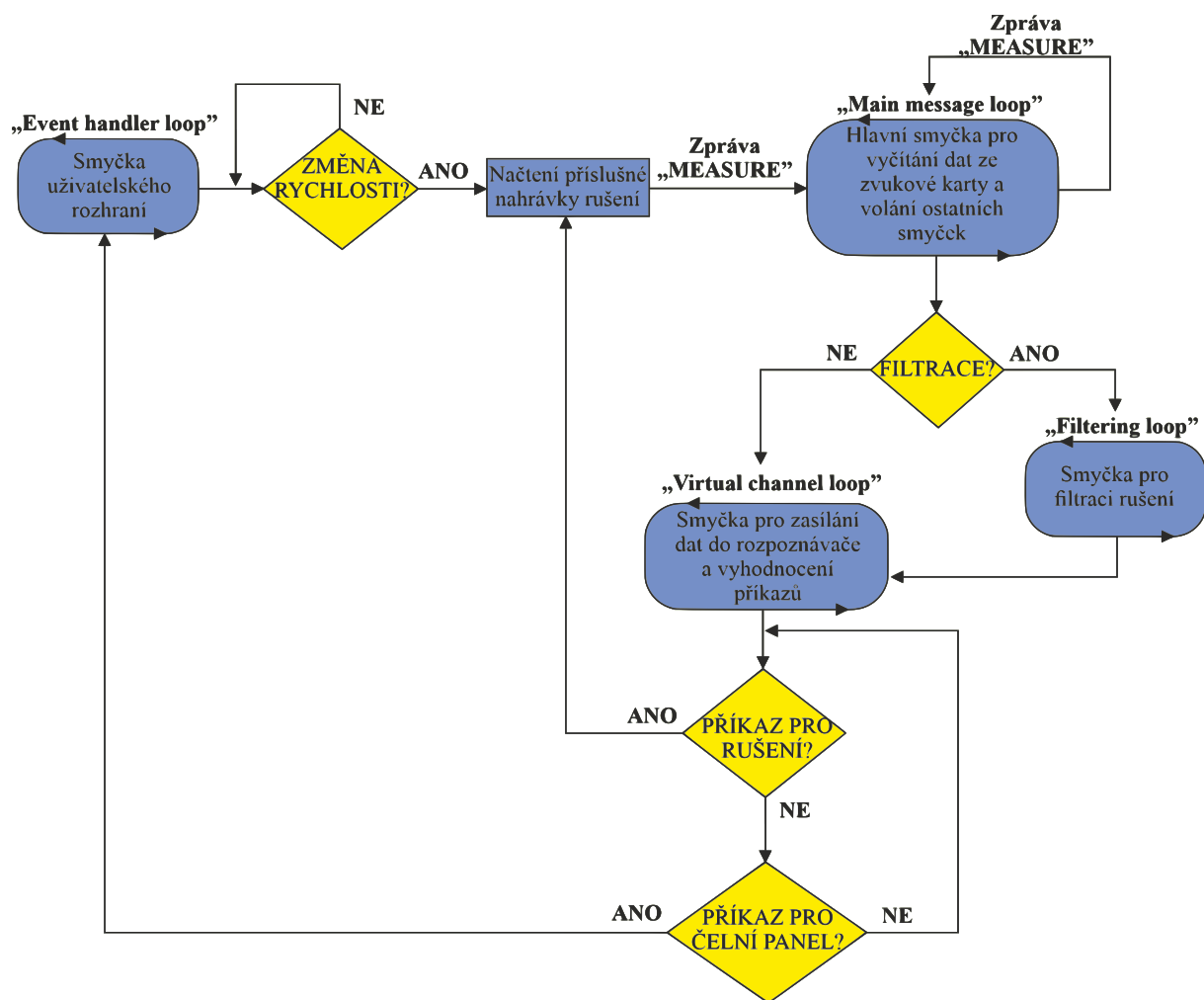
Obr. 55 Posouvání nahrávek v poli v subVI *Rotace sumu*

7.3.4 Vizualizace interiéru automobilu

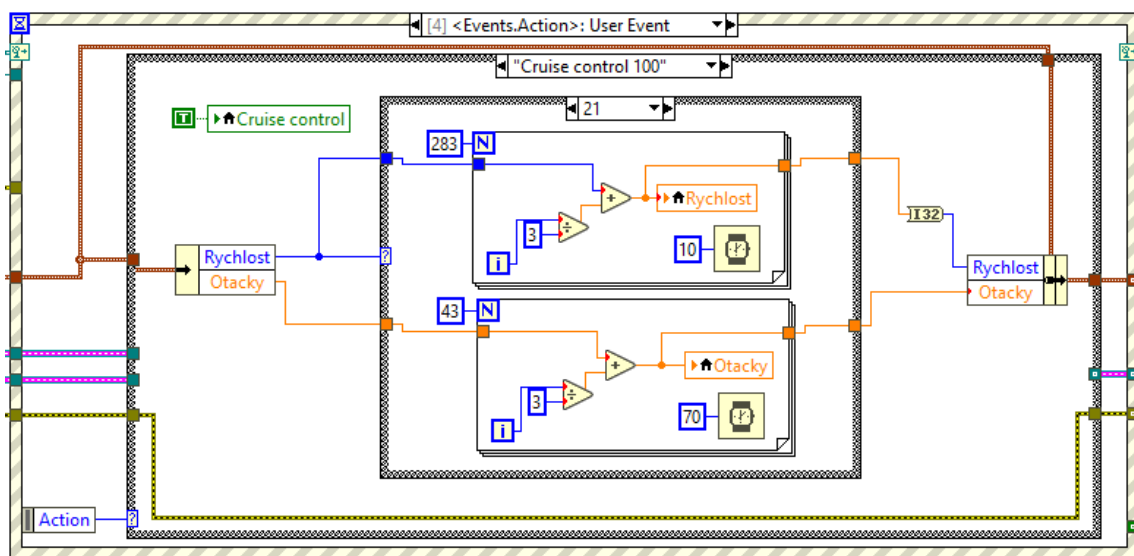
První vizualizací je interiér automobilu. Při spuštění programu proběhne inicializace front, notifikací a uživatelských akcí tzv. *User events* a je zaslána zpráva „*FIRST INIT*“ do všech smyček, kde proběhne inicializace jejich interních proměnných.

Smyčka uživatelského rozhraní obsluhuje pouze jedno tlačítko pro zapnutí/vypnutí filtrace a rychloměr pro simulaci sešlápnutí plynového pedálu. Hlavní úlohou této smyčky je tedy zpracování příkazů pro obsluhu čelního panelu přes dynamické generování *User eventu*.

Tlačítko filtrace slouží jako stupeň mezi rušenou a odfiltrovanou řečí. Pokud nebude zmáčknuto tlačítko *FILTRATION* budou data poslána do smyčky pro obsluhu virtuálního kanálu. Pokud dojde k jeho zmáčknutí, data se začnou posílat do smyčky pro filtraci, a odfiltrovaná data jsou následně zaslána do virtuálního kanálu.

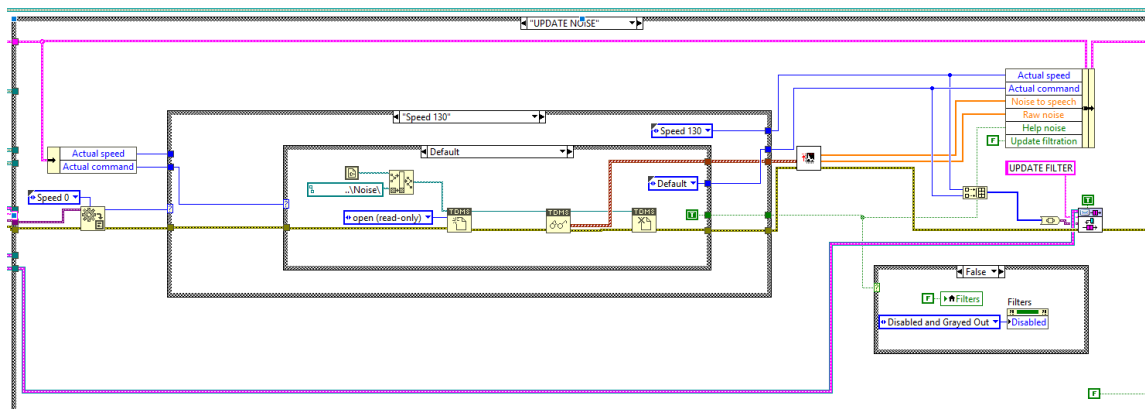


Obr. 56 Zjednodušené blokové schéma algoritmu interiéru automobilu

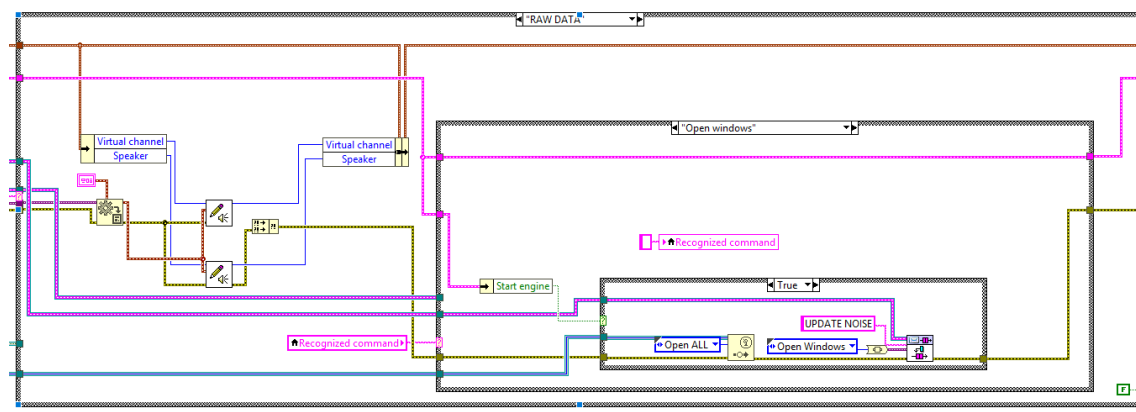


Obr. 57 Ukázka zpracování příkazu přes vygenerovaný User event

Ke stavu „Update NOISE“, který načte příslušnou nahrávku rušení lze přistoupit ze dvou smyček. Jak již bylo řečeno první z nich je smyčka uživatelského rozhraní, kde na základě pohnutí s rychloměrem, dojde k poslání zprávy a na základě zvolené rychlosti dojde k načtení příslušné nahrávky rušení. Druhá smyčka obsluhuje posílání hlasových příkazů přes virtuální kanál do rozpoznávače a následně jej vyhodnotí. Pokud tedy uživatel řekne například příkaz „Cruise control one hundred“ dojde k poslání zprávy „UPDATE NOISE“ a bude načteno rušení při 100 km/h.



Obr. 58 Vykonání stavu UPDATE NOISE



Obr. 59 Vykonání stavu pro zasílání dat do virtuálního kanálu, reproduktoru a provedení vyhodnoceného příkazu

7.3.4.1 Slovník příkazů

Jak již bylo řečeno, je potřeba definovat slovník příkazů pro rozpoznávač. Bylo definováno celkem 30 příkazů, kdy 15 příkazů slouží pro obsluhu čelního panelu a 15 příkazů pro aktivaci rušení.

Tabulka 6 Slovník příkazů pro hlasové ovládání interiéru automobilu

Příkaz	Význam
Obsluha čelního panelu	
„Start engine“	Zapnutí motoru
„Stop engine“	Vypnutí motoru
„Blinker left“	Levý blinkr
„Blinker right“	Pravý blinkr
„Daytime lights“	Denní světla
„High beam lights“	Dálková světla
„Auto lights“	Automatická dálková světla
„Front fog lights“	Přední mlhová světla
„Rear fog lights“	Zadní mlhová světla
„Accept call“	Přijetí hovoru
„Decline call“	Zamítnutí hovoru
„Radio on“	Zapnutí rádia
„Radio off“	Vypnutí rádia
„Istop on“	Zapnutí I-stop (defaultně zapnut)
„Istop off“	Vypnutí I-stop
Zapnutí/vypnutí rušení	
„Cruise control twenty“	Nastavení tempomatu na 20 km/h
„Cruise control fifty“	Nastavení tempomatu na 50 km/h
„Cruise control eighty“	Nastavení tempomatu na 80 km/h
„Cruise control one hundred“	Nastavení tempomatu na 100 km/h
„Cruise control one hundred thirty“	Nastavení tempomatu na 130 km/h
„Open windows“	Otevřít všechna okna
„Close windows“	Zavřít všechna okna
„Open the left front window“	Otevřít levé přední okno
„Open the left rear window“	Otevřít levé zadní okno
„Open the right front window“	Otevřít pravé přední okno
„Open the right rear window“	Otevřít pravé zadní okno
„Close the left front window“	Zavřít levé přední okno
„Close the left rear window“	Zavřít levé zadní okno
„Close the right front window“	Zavřít pravé přední okno
„Close the right rear window“	Zavřít pravé zadní okno

7.3.4.2 Ovládaní aplikace

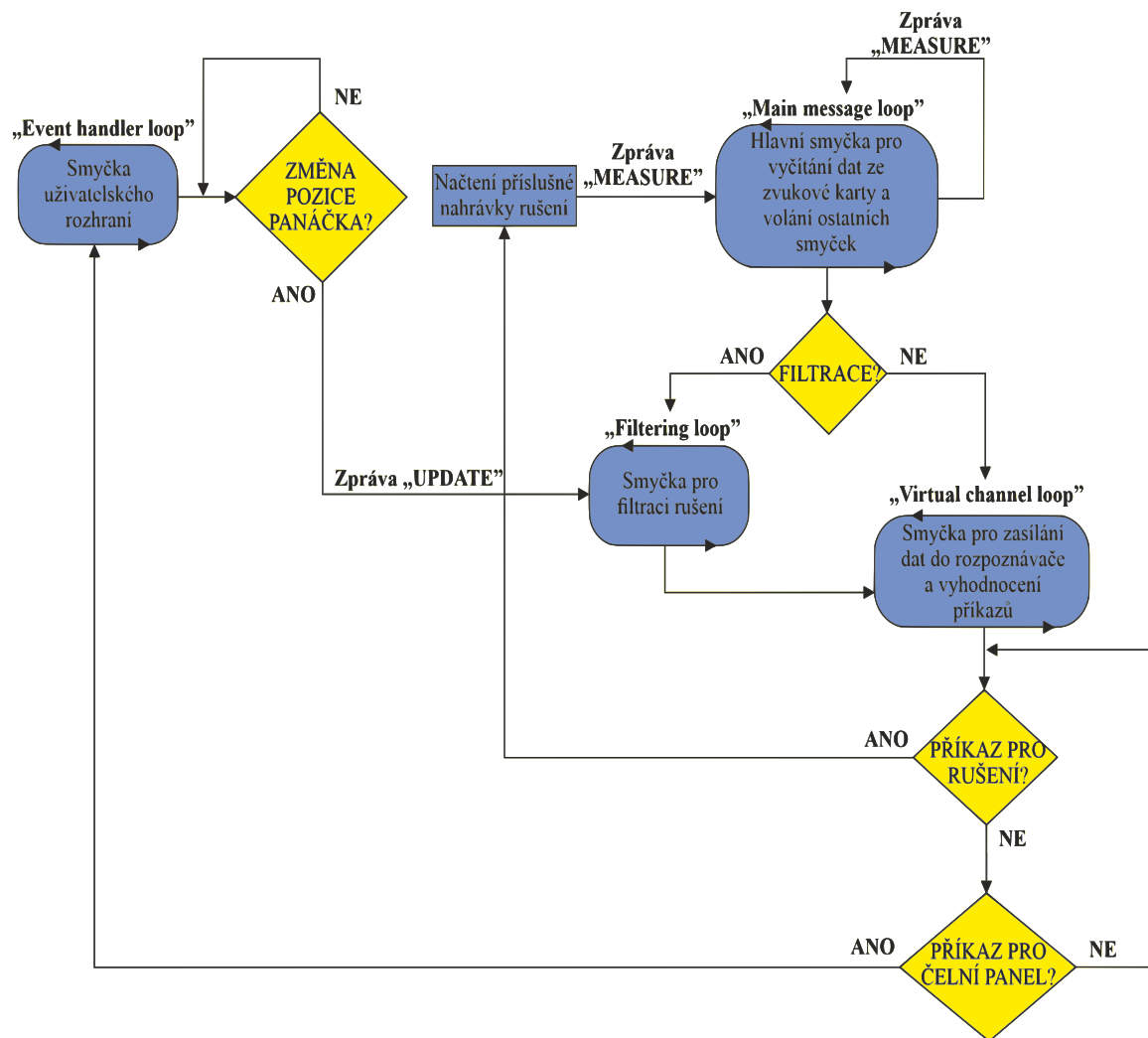
Čelní panel se skládá ze 4 hlavních indikátorů jimiž jsou otáčkoměr, rychloměr, indikátor teploty a stavu paliva. Po zapnutí aplikace je nejprve nutné říct příkaz „start engine“, kdy dojde k oživení jednotlivých budíků, je zapnuta kontrolka I-STOP a je spuštěna nahrávka rušení auta při volnoběhu. V tomto stavu uživatel může hlasově ovládat aplikaci dle definovaného slovníku. V případě vypnutí auta je nejprve nutné ručně nastavit rychlost na nulu a následně říct „stop engine“. Tím dojde k vypnutí všech indikátorů a aplikace čeká na opětovné spuštění příkazem „start engine“.



Obr. 60 Čelní panel aplikace interiéru automobilu

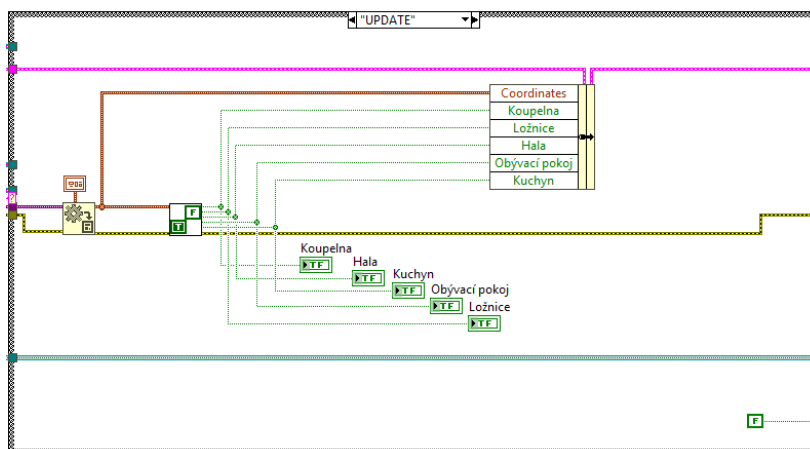
7.3.5 Vizualizace „inteligentního“ domu

Podobně jako u vizualizace interiéru automobilu je aplikace tvořena smyčkou uživatelského rozhraní, která slouží pro zapnutí/vypnutí filtrace a pro zachycení změny pozice panáčka, jehož úkolem je simulace, ve které části domu se uživatel zrovna nachází. Například pokud bude panáček umístěn do prostoru koupelny, není možné aktivovat světla nebo jiná zařízení nacházející se v jiných částech domu. Ostatní části algoritmu jsou totožné s aplikací pro interiér automobilu.



Obr. 61 Zjednodušené blokové schéma algoritmu „inteligentního“ domu

Pokud dojde ke změně pozice, je zavolána zpráva „UPDATE“ ve smyčce určené pro filtraci, kde jsou přenesena data o aktuálních souřadnicích panáčka a je vyhodnoceno, ve které místnosti se nachází. Následně jsou data za psána do proměnných a uložena do klastru.



Obr. 62 Vyhodnocení souřadnic a jejich zápis do proměnných a klastru

7.3.5.1 Slovník příkazů

Slovník pro „inteligentní“ domácnost je definován příkazy pro zapnutí/vypnutí světel a zatažení/vytažení rolet, které lze použít ve všech místnostech. Ostatní příkazy lze pak použít v rámci dané místnosti, viz Tabulka 7.

Tabulka 7 Slovník příkazů pro hlasové ovládání "inteligentní" domácnosti

Příkaz	Význam	Místnost
„Light on“	Zapnout světlo	Všechny
„Light off“	Vypnout světlo	Všechny
„Turn on the washing machine“	Zapnout pračku	koupelna
„Turn off the washing machine“	Vypnout pračku	koupelna
„Dimm up“	Zesílit jas světla (lampička)	ložnice
„Dimm down“	Ztlumit jas světla (lampička)	ložnice
„Turn on the vacuum cleaner“	Zapnout vysavač	ložnice
„Turn off the vacuum cleaner“	Vypnout vysavač	ložnice
„Turn on the dishwasher“	Zapnout myčku	kuchyň
„Turn off the dishwasher“	Vypnout myčku	kuchyň
„Fan on“	Zapnout ventilátor	Hala
„Fan off“	Vypnout ventilátor	Hala
Turn on the TV	Zapnout televizi	Obývací pokoj
Turn off the TV	Vypnout televizi	Obývací pokoj
„Blinds up“	Všechny rolety nahoru	Kuchyň/hala/obývací pokoj/ložnice
„Blinds down“	Všechny rolety dolů	Kuchyň/hala/obývací pokoj/ložnice
„Blinds up left“	levá roleta nahoru	Kuchyň/hala/obývací pokoj
„Blinds up right“	pravá roleta nahoru	Kuchyň/hala/obývací pokoj
„Blinds up middle“	prostřední roleta nahoru	Kuchyň/hala/obývací pokoj
„Blinds down left“	levá roleta dolů	Kuchyň/hala/obývací pokoj
„Blinds down right“	pravá roleta dolů	Kuchyň/hala/obývací pokoj
„Blinds down left“	levá roleta dolů	Kuchyň/hala/obývací pokoj
„Blinds down middle“	prostřední roleta dolů	Kuchyň/hala/obývací pokoj

7.3.5.2 Ovládání aplikace

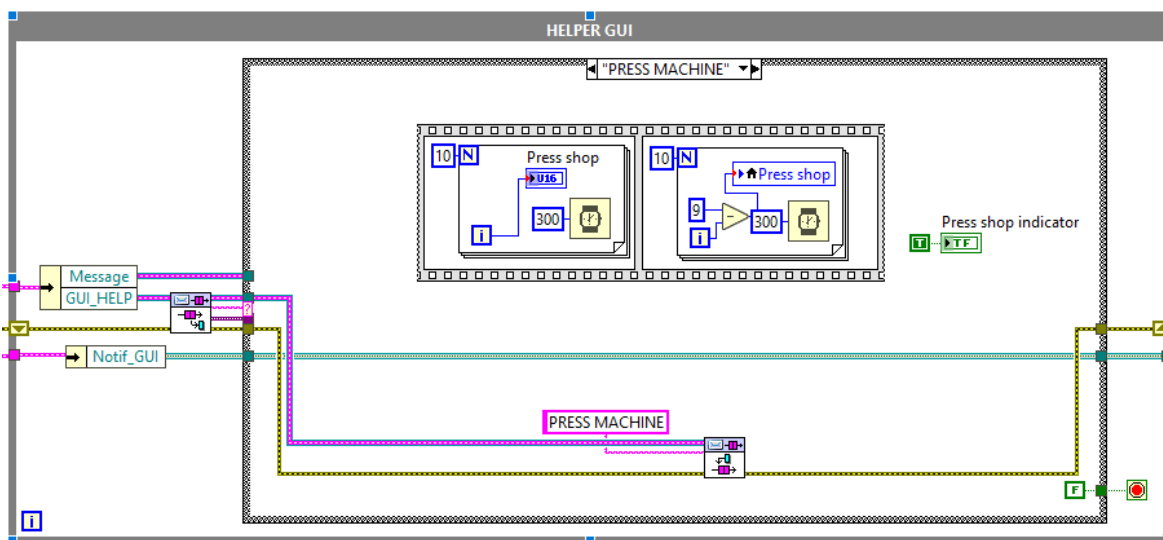
Aplikace „inteligentní“ domácnosti sestává z 5 místností, konkrétně koupelny, ložnice, haly, obývacího pokoje a kuchyně, ve kterých lze udělovat hlasové příkazy. Po zapnutí aplikace je panáček umístěn do výchozí pozice, která se nachází v předsíni. Tento stav je výchozí, je tedy možné udělovat hlasové příkazy ale bez jejich odezvy. Pro hlasové řízení je vždy potřeba umístit panáčka do předem zvolené pozice, kdy na základě vyhodnocení souřadnic je proved příkaz. Například pokud bude chtít uživatel zatahnout rolety v ložnici, musí panáčka přemístit do této pozice, viz Obr. 63. Tak jsou ošetřeny nežádoucí stavy, kdy uživatel přemístí panáčka například do kuchyně a bude chtít ovládat zařízení v ostatních částech domu.



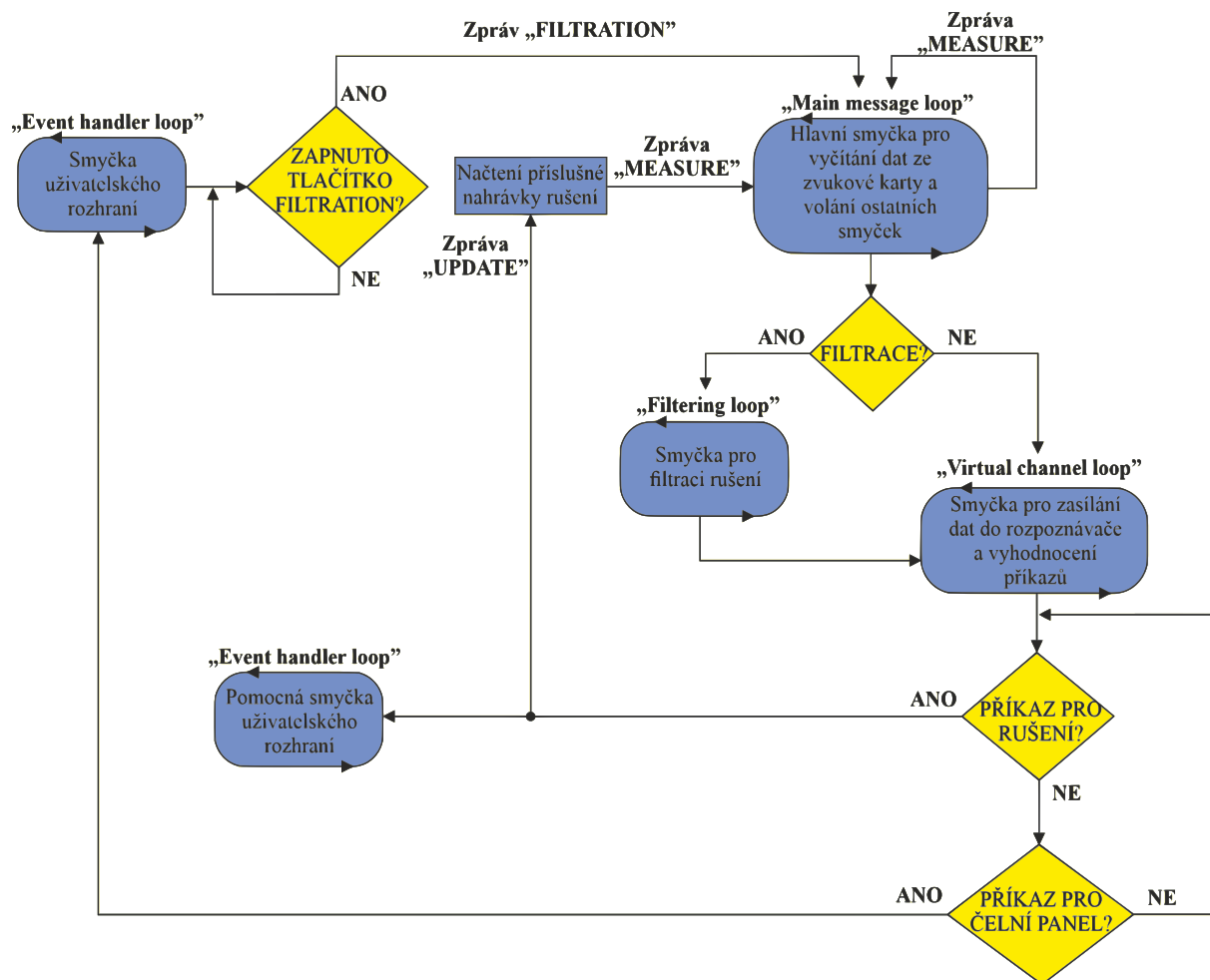
Obr. 63 Čelní panel aplikace "inteligentní" domácnosti

7.3.6 Vizualizace výrobní linky

Poslední vizualizací je výrobní linka, která je tvořena 5 smyčkami. Jedná se o stejně naprogramovanou architekturu jako u předchozích vizualizací s tím rozdílem, že byla implementována pomocná smyčka pro uživatelské rozhraní. Tato smyčka slouží pro vizualizaci laserové/magnetické svářečky a lisu.



Obr. 64 Ukázka aktivace lisu v pomocné smyčce



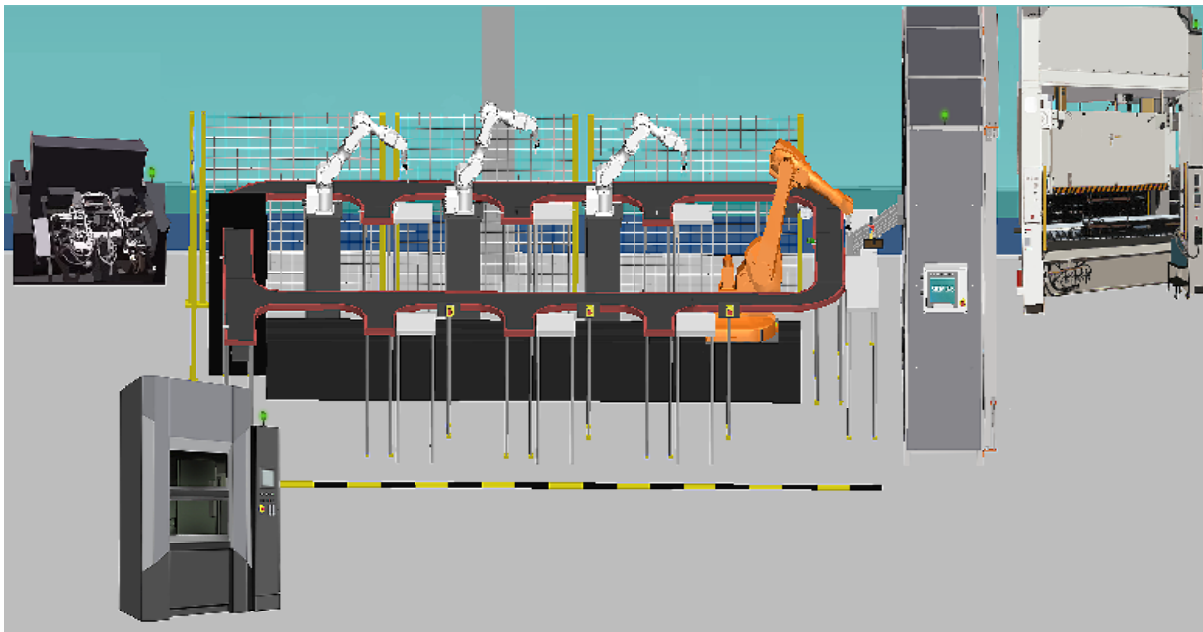
7.3.6.1 Slovník příkazů

Tabulka 8 Slovník příkazů pro hlasové ovládání výrobní linky

Příkaz	Význam
Obsluha čelního panelu	
„Turn on the production line“	Zapnutí výrobní linky
„Turn off the production line“	Vypnutí výrobní linky
„Homing“	Nastavení ramen do výchozí pozice
„Turn on the main arm“	Aktivace hlavního, oranžového ramene
„Turn on the right arm“	Aktivace pravého ramena
„Turn on the middle arm“	Aktivace prostředního ramena
„Turn on the left arm“	Aktivace levého ramena
„Turn on the belt“	Zapnutí pásu
„Move to the next position“	Posunutí na další pozici
„Move to the end position“	Posunutí do koncové pozice
Zapnutí/vypnutí rušení	
„Turn on the laser welder“	Zapnutí laserové svářečky
„Turn off the laser welder“	Vypnutí laserové svářečky
„Turn on the mag welder“	Zapnutí magnetické svářečky
„Turn off the mag welder“	Vypnutí magnetické svářečky
„Turn on the press machine“	Zapnutí lisu
„Turn off the press machine“	Vypnutí lisu

7.3.6.2 Ovládání aplikace

Čelní panel výrobní linky **Chyba! Nenalezen zdroj odkazů.**, se skládá z hlavního rozvaděče, čtyř robotických ramen, dopravníku, laserové/magnetické svářečky a lisu. Po zapnutí aplikace je nejprve nutné říct příkaz „Turn on the production line“ pro aktivaci rozvaděče. Protože se jedná o automatickou linku, která byla upravena pro potřeby hlasového ovládání byla zavedena režie. To znamená, že je nejprve nutné aktivovat buďto hlavní rameno (oranžové), které postupně naskládá jednotlivé komponenty na desku, které následně přenesou k pravému ramenu nebo aktivovat pás, který z výchozí pozice přiveze desku na pozici pod pravým ramenem. Oba tyto příkazy musí být provedeny jinak nelze aktivovat pravé rameno. Jakmile pravé rameno postaví z komponent domeček, je možné desku posunout na pozici levého ramene příkazem „Move to the next position“. Domeček je tak pomocí pásu převezen na pozici pod levým ramenem, které testuje, jestli je mezi kostkami otvor pro implementaci dveří. Jakmile tento úkon vykoná je možné převést domeček na koncovou pozici příkazem „Move to the end position“. V koncovém stavu setrvá domeček 2 sekundy a následně je sekvence vyresetována, kdy deska je opět umístěna do výchozí pozice, a uživatel má tak možnost celý proces opakovat.

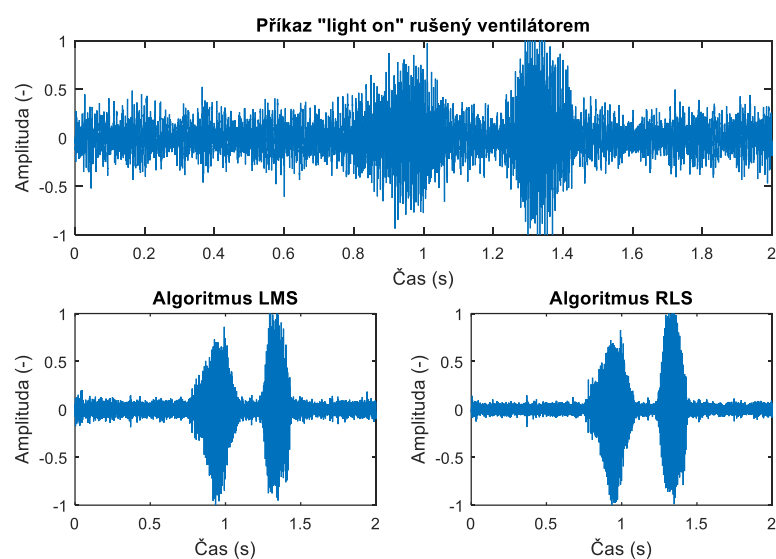


Obr. 66 Čelní panel aplikace výrobní linky

7.4 Vybrané metody filtrace a úspěšnost rozpoznání

Pro potlačení rušení byla vybrána metoda ICA spolu adaptivními metodami, jmenovitě algoritmem LMS a RLS.

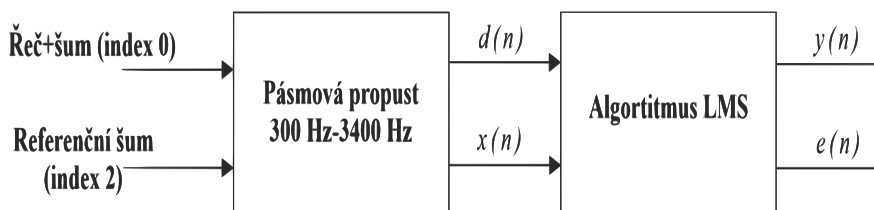
Algoritmus LMS i přes jeho jednoduchost a matematickou nenáročnost vykazoval podobné výsledky globálního SNR jako algoritmus RLS (zhruba o 2 dB horší). Naopak algoritmus RLS sice vykazoval lepší výsledky filtrace ale za cenu mnohonásobného zvýšení výpočetní doby, které průměrně činily 15 sekund. Tímto se stává algoritmus RLS nepoužitelným v aplikacích reálného času. Proto bylo od této metody upuštěno a testován byl pouze algoritmus LMS.



Obr. 67 Ukázka filtrace algoritmu LMS a RLS

7.4.1 Hledání optimálního nastavení parametrů algoritmu LMS

Jelikož není ve vizualizacích možné předem určit jaký bude následující příkaz, a tak nastavit nejideálnější parametry algoritmu LMS bylo nejprve nutné provést offline identifikaci. Ta byla provedena tak, že pro každý příkaz a rušení byly vyhledány optimální hodnoty podle globálního SNR. Z těchto nalezených hodnot byla následně vybrána nejlepší délka filtru M a konvergenční konstanta μ . Filtrace probíhá ve dvou krocích (Obr. 68), kdy je nejprve přiveden řečový signál kontaminovaný rušením a referenční šum do filtru typu pásmové propusti nastavený na hodnotu 300 Hz – 3400 Hz, což je frekvenční pásmo odpovídající lidské řeči. Poté jsou odfiltrované signály poslány do algoritmu LMS, kdy $y(n)$ je odfiltrovaný signál, a $e(n)$ chyba filtrace.



Obr. 68 Měřicí řetězec filtrace pro algoritmus LMS

Z Tabulka 9, Tabulka 10, Tabulka 11 vyplývá, že při vzrůstající energii rušení budou větší požadavky na adaptivní filtr. To znamená vyšší délka filtru M a konvergenční konstanta μ . Při testování se filtr jevil tak, že při vzrůstající energii rušení, potřeboval vyšší délku filtrace, avšak zde nastává problém, kdy při vysokých hodnotách (řádově délka filtru 1000 a vyšší), dochází ke zkreslení užitečného signálu, který je částečně odfiltrován a roste chyba filtrace. To samé platí u konvergenční konstanty, kdy při vysokých hodnotách (nad 0,1) dochází k rozvažování vah a filtr se tak stává nestabilní. Další problém je výpočetní čas. Čím delší bude délka filtru a čím menší bude konvergenční konstanta, tím bude výpočet trvat déle a opačně. Tak nastává protichůdná situace, kdy je snaha o co nejlepší filtrace v minimálním čase.

Tabulka 9 Optimální nastavení parametrů pro algoritmus LMS, vizualizaci interiéru automobilu

LMS	Rušení					
	Zavřená okna		Otevřená levé přední okno		Otevřená okna	
Rychlost	Délka filtru M	Konverg. konst. μ [-]	Délka filtru M	Konverg. konst. μ [-]	Délka filtru M	Konverg. konst. μ [-]
80 km/h	980	0,001	530	0,01	880	0,1
100 km/h	1230	0,001	1230	0,01	190	0,1
130 km/h	1120	0,001	1220	0,01	400	0,1

Tabulka 10 Optimální nastavení parametrů pro algoritmus LMS, vizualizaci "inteligentní" domácnosti

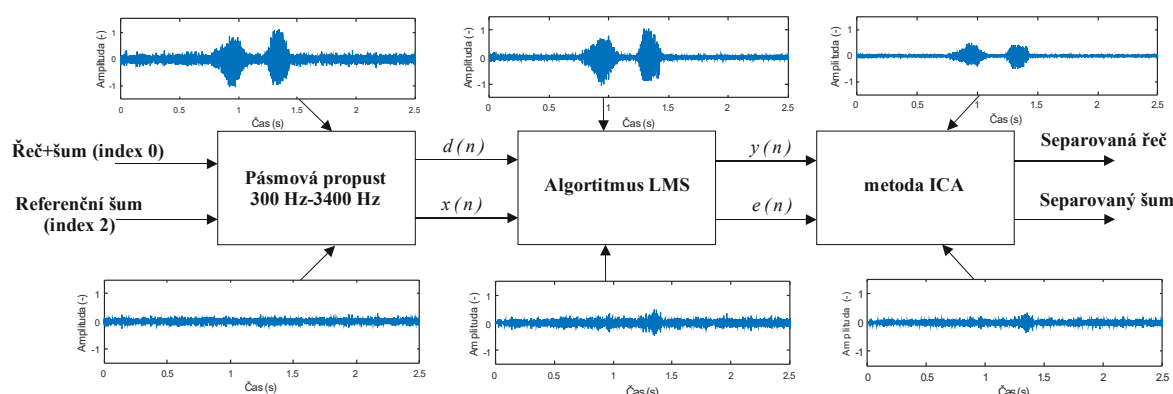
Rušení	Délka filtru M	Konverg. konstanta μ [-]
Pračka	240	0,01
Vysavač	80	0,001
Ventilátor	210	0,01
Myčka	40	0,01
Televize	110	0,01

Tabulka 11 Optimální nastavení parametrů pro algoritmus LMS, vizualizaci "výrobní linky"

Rušení	Délka filtru M	Konverg. konstanta μ [-]
Laserová svářečka	180	0,001
Magnetická svářečka	910	0,001
Lis	880	0,01

7.4.2 Analýza nezávislých komponent

Pro hlasové ovládání vizualizací byl použit pouze jeden mikrofón, viz kapitola 7.3.3, a proto není možné řešit klasický „cocktail party problém“. Z tohoto důvodu byla použita hybridní filtrace (Obr. 69), kdy za výstup adaptivního filtru byla implementována metoda ICA. Po průchodu pásmovou propustí jsou signály zaslány do algoritmu LMS a odfiltrovány. Z průběhů je patrné, že algoritmus značně potlačuje rušení, avšak současně roste chyba filtrace. To je způsobeno snahou algoritmu LMS co nejvíce potlačit rušení, kdy částečně filtruje i řeč. To je jedna z vlastností adaptivních algoritmů, s kterou je třeba počítat. V LabVIEW byla použita funkce v knihovně *Signal Processing* → *Time Series Analysis*.



Obr. 69 Měřicí řetězec hybridní filtrace LMS a ICA

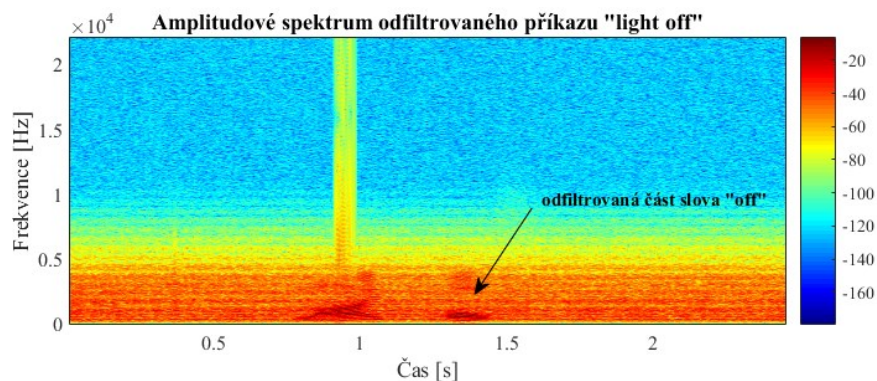
Tabulka 12 Nastavení parametrů funkčního bloku ICA

Parametr	Hodnota
Metoda	Fast ICA
Počet komponent	2
Počet iterací	1000
Konvergenční tolerance	0,000001

7.4.3 Úspěšnost rozpoznání

Pro každý příkaz bylo provedeno sto opakování, kdy na základě stavu rozpoznal/nerozpoznal byla vyhodnocena úspěšnost rozpoznání. Příkazy byly udělovány do mikrofónu v konstantní vzdálenosti 15 cm. Při testování výrobní linky, především rušení lisu, musel být mikrofón těsně u úst z důvodu velké energie signálů (zvuková karta místy ořezávala signál kvůli překročení maximálního rozlišení). Rozpoznávač vykazoval nejmenší míru rozpoznání pro slova končící na „off“ („light off“, „i-stop off“,

„radio off“). To může být zapříčiněno fonetickou stránkou daného příkazu, který má nízkou energii. Další příčinou je vlastnost adaptivních filtrů, kdy je dané slovo potlačeno (mírně filtrováno). Ukázka odfiltrovaného příkazu „light off“ je na Obr. 70.



Obr. 70 Spektrogram odfiltrovaného slova "off"

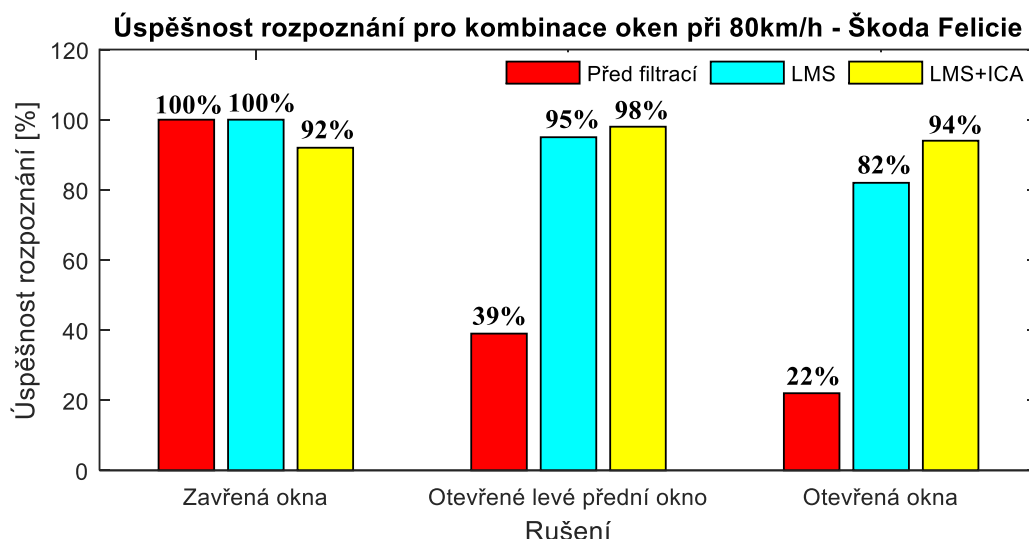
7.4.3.1 Interiér automobilu

Z Tabulka 13 a Tabulka 14 lze vypožorovat, že při zavřených oknech byla úspěšnost před filtrací 100 %. Při testování došlo k fenoménu známém především v telekomunikační a sdělovací technice, kdy užitečný signál vykazoval mírné zhoršení rozpoznání po filtraci. Tento jev je běžný především u metody RASTA-PLP (kapitola 4.2.2), která zhoršuje přesnost rozpoznávání v situacích, kdy je na vstupu klasifikátoru čistá, nezkreslená řeč.

Tabulka 13 Výsledky úspěšnosti rozpoznání pro automobil Škoda Felicie, 80 km/h

LMS a ICA	80 km/h Škoda Felicie								
	Zavřená okna			Otevřená levé přední okno			Otevřená okna		
Příkaz	PŘED [%]	LMS [%]	LMS+ ICA [%]	PŘED [%]	LMS [%]	LMS+ ICA [%]	PŘED [%]	LMS [%]	LMS+ ICA [%]
„Blinker left“	100	100	92	33	100	100	0	85	78
„Blinker right“	100	100	90	42	100	100	0	91	80
„Daytime lights“	100	100	100	54	100	100	15	100	100
„Hig. beam lights“	100	100	100	42	100	100	60	100	100
„Auto lights“	100	100	100	36	94	100	18	100	100
„Cru. con. 20“	100	100	100	60	98	100	9	63	92
„Cru. con. 50“	100	100	100	45	100	100	9	81	100
„Cru. con. 100“	100	100	100	54	97	100	6	100	98
„Cru. con. 130“	100	100	100	39	100	100	5	91	96
„Front fog light“	100	100	100	12	100	100	12	100	100
„Rear fog lights“	100	100	100	36	100	100	79	98	100
„Accept call“	100	100	100	27	57	100	42	44	100
„Decline call“	100	100	100	75	90	100	33	57	100
„Radion on“	100	100	100	33	100	100	15	42	94
„Radio off“	100	100	42	12	100	78	33	85	65
„I-stop on“	100	100	100	5	84	100	6	63	100
„I-stop off“	100	100	38	66	100	85	39	97	93

Z grafu na Obr. 71 byla úspěšnost rozpoznání před filtrací při levém otevřeném okně 39 % a pro otevřená okna 22 %. Pro adaptivní algoritmus LMS byla při levém otevřeném okně úspěšnost 98 %, avšak například u příkazu „accept call“ byla úspěšnost pouze 57 %. Metoda ICA vykazovala o 3% vyšší úspěšnost oproti algoritmu LMS, ale například u příkazu „radio off“ měla jen 78 % úspěšnost, což je o 22% horší výsledek než u algoritmus LMS. Při otevřených oknech jednoznačně dominuje metoda ICA, která měla celkově o 12% vyšší úspěšnost. Opět u příkazu „radio off“ a „i-stop off“ vykazovala horší výsledky.

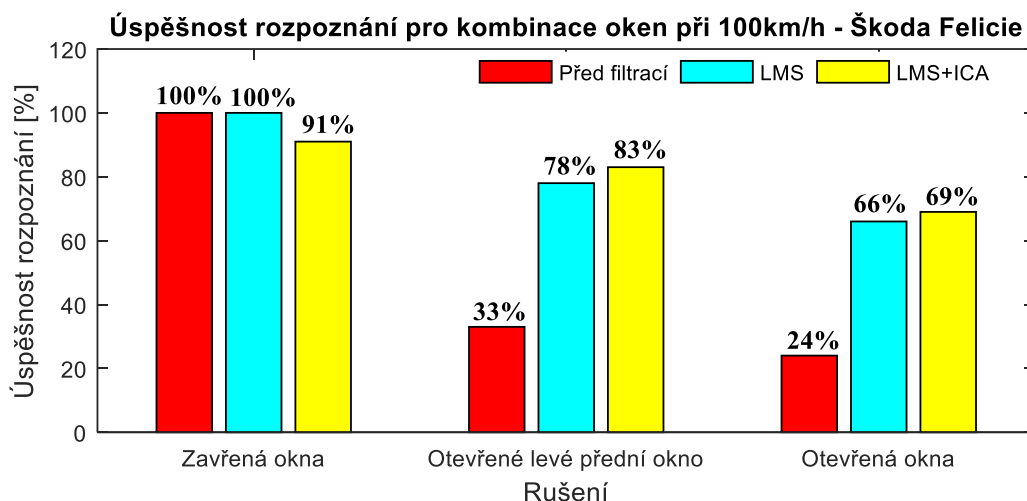


Obr. 71 Výsledky rozpoznání pro automobil Škoda Felicie, 80 km/h

Tabulka 14 Výsledky úspěšnosti rozpoznání pro automobil Škoda Felicie, 100 km/h

LMS a ICA	100 km/h Škoda Felicie								
	Zavřená okna			Otevřeně levé přední okno			Otevřená okna		
Příkaz	PŘED [%]	LMS [%]	LMS+ ICA [%]	PŘED [%]	LMS [%]	LMS+ ICA [%]	PŘED [%]	LMS [%]	LMS+ ICA [%]
„Blinker left“	100	100	96	24	84	9	0	72	12
„Blinker right“	100	100	92	36	78	3	0	35	8
„Daytime lights“	100	100	100	42	92	88	27	90	78
„Hig. beam lights“	100	100	100	50	93	84	12	100	51
„Auto lights“	100	100	100	21	100	90	18	72	75
„Cru. con. 20“	100	100	100	33	60	51	0	36	42
„Cru. con. 50“	100	100	100	54	88	72	9	66	36
„Cru. con. 80“	100	100	100	24	63	69	0	60	54
„Cru. con. 130“	100	100	100	18	42	31	0	35	3
„Front fog light“	100	100	100	21	69	80	18	72	75
„Rear fog lights“	100	100	100	36	58	90	78	54	87
„Accept call“	100	100	100	18	100	95	39	100	73
„Decline call“	100	100	100	34	100	100	57	100	72
„Radion on“	100	100	100	33	100	78	27	75	70
„Radio off“	100	100	38	8	9	66	42	82	54
„I-stop on“	100	100	100	5	100	87	9	81	66
„I-stop off“	100	100	32	52	6	90	63	28	87

Při 100 km/h a otevřeném levém okně opět vykazuje metoda ICA lepší filtraci, a to přibližně o 5 %. Při testování příkazů „*blinker left*“ ale měla úspěšnost jen 9 %, kdy algoritmus LMS vykazoval úspěšnost 84 %. Podobně u příkazu „*blinker right*“ je výsledek pro metodu ICA pouze 3 % a pro adaptivní algoritmus 78 %. Při otevřených oknech jde vidět, že se algoritmy postupně dostávají na srovnatelné výsledky, kdy metoda ICA měla pouze o 3 % vyšší úspěšnost. To může být zapříčiněno charakterem rušení, kdy z naměřených průběhu (Obr. 46) jde vidět, že se jedná spíše o tlakové vlny způsobené porvy větru.



Obr. 72 Výsledky rozpoznání pro automobil Škoda Felicie, 100 km/h

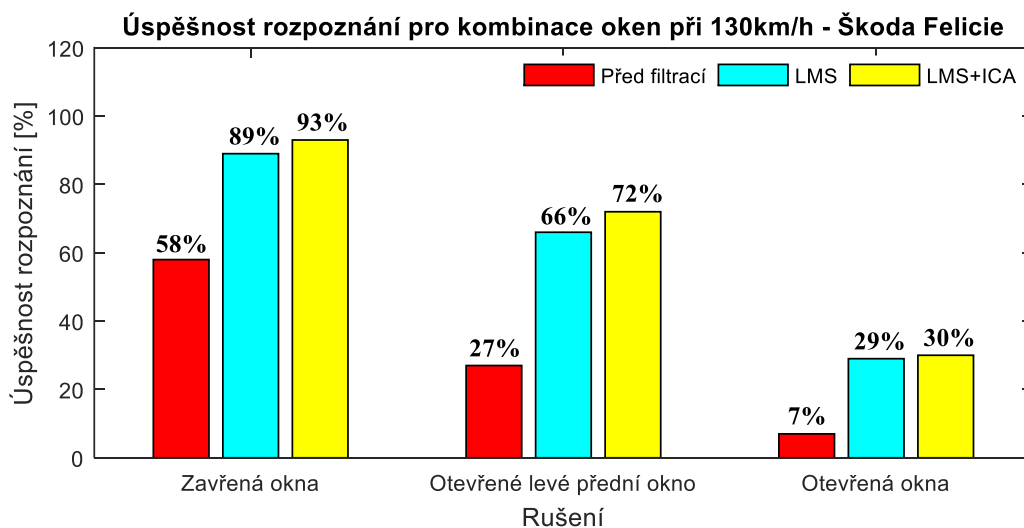
Tabulka 15 Výsledky úspěšnosti rozpoznání pro automobil Škoda Felicie, 130 km/h

LMS a ICA	130 km/h Škoda Felicie								
	Zavřená okna			Otevřené levé přední okno			Otevřená okna		
Příkaz	PŘED [%]	LMS [%]	LMS+ ICA [%]	PŘED [%]	LMS [%]	LMS+ ICA [%]	PŘED [%]	LMS [%]	LMS+ ICA [%]
„Blinker left“	9	78	93	60	81	60	0	12	52
„Blinker right“	30	90	57	27	63	72	0	15	66
„Daytime lights“	12	90	100	60	78	98	0	75	45
„Hig. beam lights“	57	100	100	5	63	100	0	73	30
„Auto lights“	45	93	100	0	57	90	0	65	27
„Cru. con. 20“	100	93	92	0	60	90	0	0	12
„Cru. con. 50“	100	96	100	0	33	75	0	0	18
„Cru. con. 80“	100	100	100	0	66	72	0	0	26
„Cru. con. 100“	100	96	100	6	69	57	0	0	21
„Front fog light“	57	81	100	0	30	60	0	51	40
„Rear fog lights“	63	100	100	28	69	88	4	40	35
„Accept call“	95	51	100	100	95	100	1	27	15
„Decline call“	100	93	100	60	84	96	0	10	12
„Radion on“	0	81	100	0	88	30	0	0	5
„Radio off“	50	69	60	5	38	36	45	30	10
„I-stop on“	0	100	100	0	51	40	0	0	0
„I-stop off“	50	94	78	100	100	65	63	100	82

Pro poslední měření automobilu při 130 km/h lze pozorovat (Obr. 73), že i při zavřených oknech má rozpoznávač před filtrací pouze 58 % úspěšnost. Metoda adaptivního algoritmu LMS vykazoval 89 % úspěšnost a metoda ICA 93 %. Opět největší problém byl zaznamenán u příkazu „radio off“, kdy před filtrací byla úspěšnost 50 % a po filtraci algoritmem LMS 69 % a metodou ICA pouhých 5 %.

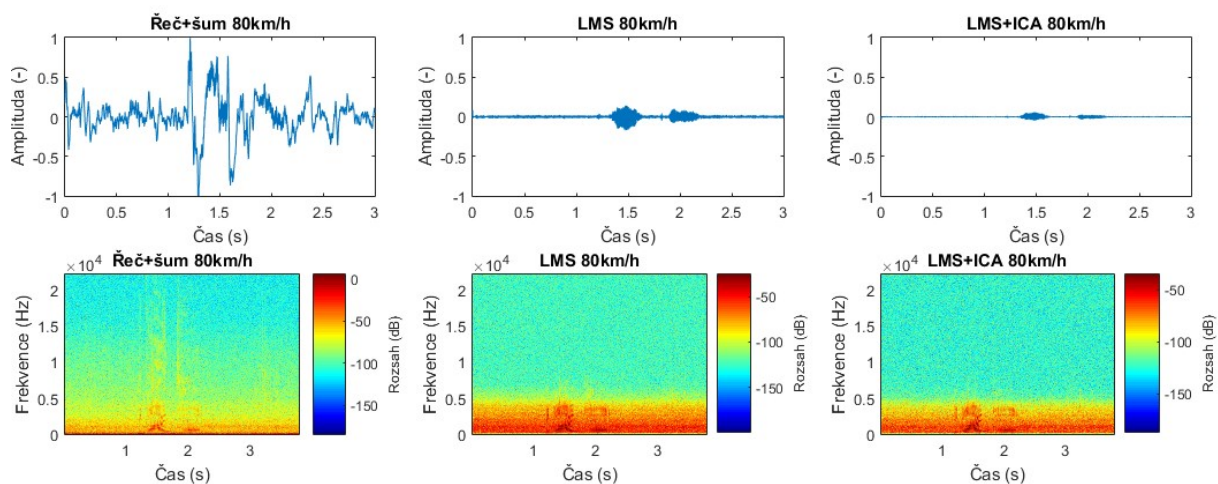
Při otevřeném levém okně klesla úspěšnost rozpoznání na 27 %, kdy byl rozdíl mezi metodami 6 %. Zde rozpoznávač nedokázal nerozeznat příkazy „auto lights“, „cruise control twenty“, „cruise control fifty“ a „cruise control eighty“. Po filtraci byla úspěšnost těchto příkazů pro algoritmus LMS 57 %, 60 %, 33 % a 66 %. Metoda ICA dosahoval zhruba o 30% lepší úspěšnost pro příkazy „auto lights“ a „cruise control twenty“, tedy 90 %. Pro příkaz „cruise control fifty“ dokonce 42 % a pro „cruise control eighty“ měla o 6% vyšší úspěšnost. Naopak pro příkaz „cruise control one hundred“ vykazoval ICA o 12% horší rozpoznání než algoritmus LMS.

Pro otevřená okna jde vidět pouze 30% úspěšnost. Tento stav je ale již velice krajní. Jak bylo řečeno, při otevřených oknech docházelo k tlakovým vlnám, kde je řeč zastíněna tímto rušením.

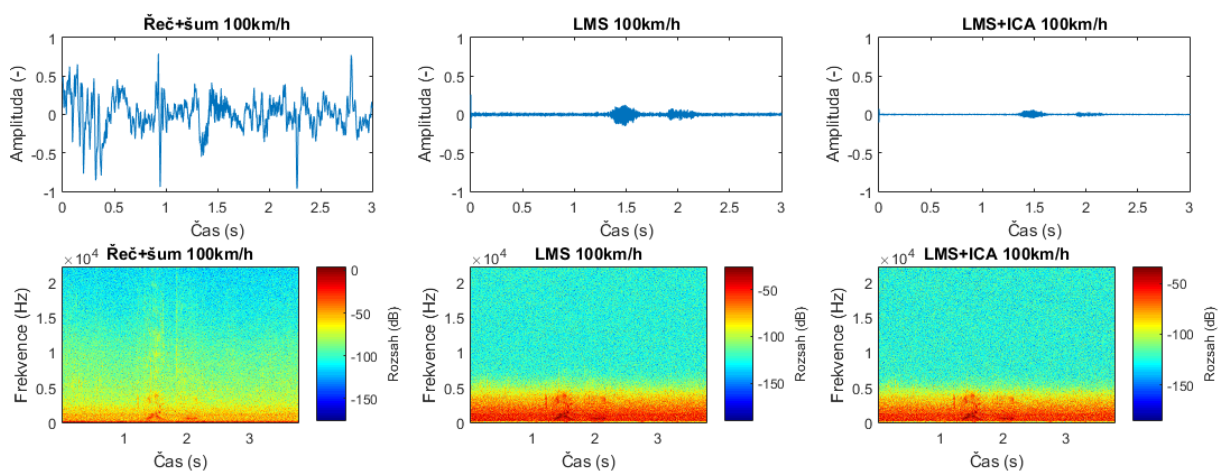


Obr. 73 Výsledky rozpoznání pro automobil Škoda Felicie, 130 km/h

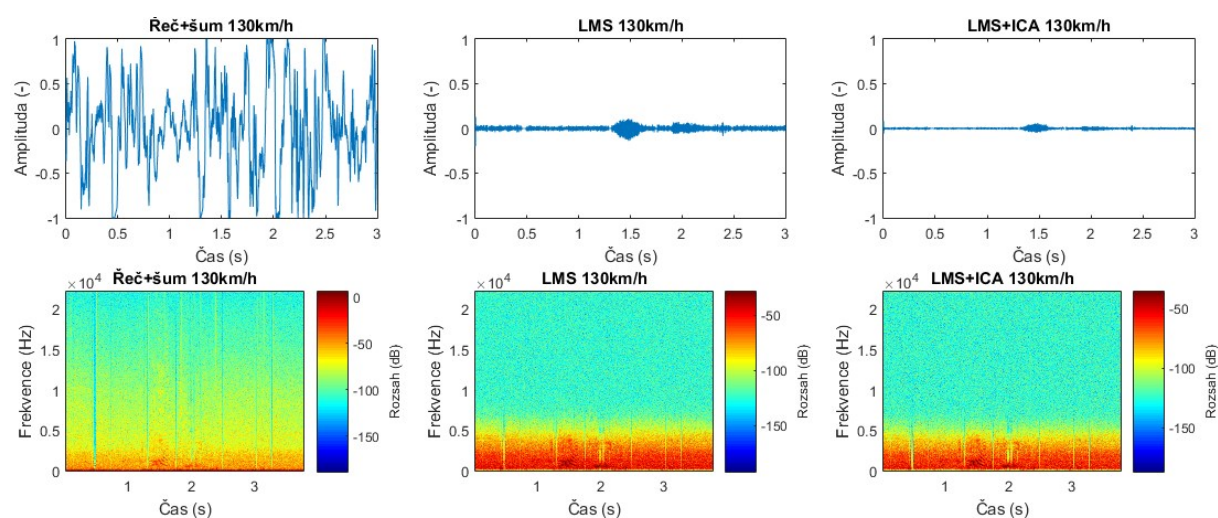
Ze spektrogramů (Obr. 74, Obr. 75, Obr. 76) lze vidět, že kombinace LMS+ICA je po filtraci nepatrně lepší než LMS v případech 80 a 100 km/h, kdy při rychlostech 130 km/h lze pozorovat pulzy, což jsou zmíněné tlakové vlny. V tomto stavu je užitečný signál jen nepatrně odfiltrován. Pro tento případ byla spíše testována robustnost algoritmů a zjištění jejich chování v daném prostředí.



Obr. 74 Porovnání spektrogramů při zavřených oknech, příkaz „decline call“, 80 km/h



Obr. 75 Porovnání spektrogramů při zavřených oknech, příkaz „decline call“, 100 km/h



Obr. 76 Porovnání spektrogramů při zavřených oknech, příkaz „decline call“, 130 km/h

7.4.3.2 „Inteligentní domácnost“

Pro rušení pračky byly testovány 3 příkazy: „*Light on*“, „*Light off*“, „*Turn off the wasching machine*“. Z Tabulky 16 lze vidět, že před filtrací byla úspěšnost 28 %, 21 % a 85 %. U vypnutí pračky je velká úspěšnost filtrace způsobena především tím, že samotný rozpoznávač má učící se algoritmus, kdy vracel předešlé hodnoty, což lze vidět na výsledcích filtrace, kde byla úspěšnost rozpoznání horší.

Tabulka 16 Výsledky úspěšnosti rozpoznání pro rušení pračky

LMS a ICA	Pračka maximální hlasitost		
Příkaz	Před [%]	LMS [%]	LMS+ICA [%]
„ <i>Light on</i> “	28	100	100
„ <i>Light off</i> “	21	45	70
„ <i>Turn off the wasching machine</i> “	85	60	78

Pro rušení vysavačem (Tabulka 17) byla průměrná úspěšnost před filtrací pouze 1 %, kdy byl rozeznán pouze příkaz „*dimm up*“ z 5 % a „*dimm down*“ ze 3 %. Po filtraci byla průměrná úspěšnost pro algoritmus LMS 80 % a pro metodu ICA 86 %. Nejméně úspěšnost byla zaznamenána u příkazu „*ligh off*“, kdy filtrace pro algoritmus byla pouze 27 % a pro metodu ICA 45 %.

Tabulka 17 Výsledky úspěšnosti rozpoznání pro rušení vysavače

LMS a ICA	Vysavač maximální hlasitost		
Příkaz	Před [%]	LMS [%]	LMS+ICA [%]
„ <i>light on</i> “	0	100	100
„ <i>light off</i> “	0	27	45
„ <i>blinds down</i> “	0	95	100
„ <i>blinds up</i> “	0	93	100
„ <i>Dimm up</i> “	5	100	100
„ <i>dimm down</i> “	3	100	100
„ <i>Turn off the vacuum cleaner</i> “	0	45	60

1

Pro rušení ventilátorem (Tabulka 18) byla průměrná úspěšnost před filtrací pouze 11 %, kdy byly nejvíce rozeznány příkazy „*light on*“ 42 % a „*light off*“ 24 %. Po filtraci byla průměrná úspěšnost pro algoritmus LMS 82 % a pro metodu ICA 91 %. Nejméně úspěšnost byla zaznamenána opět u příkazu „*fan off*“, kdy filtrace pro algoritmus byla pouze 18 % a pro metodu ICA 13 %.

Tabulka 18 Výsledky úspěšnosti rozpoznání pro rušení ventilátoru

LMS a ICA	Ventilátor maximální hlasitost		
	Před [%]	LMS [%]	LMS+ICA [%]
„light on“	42	100	100
„light off“	24	28	90
„blinds down“	2	100	100
„blinds up“	5	91	100
„blinds down left“	3	88	100
„blinds down right“	0	83	100
„blinds down middle“	0	100	100
„blinds up left“	6	95	100
„blinds up right“	9	100	100
„blinds up middle“	15	100	100
„Fan off“	18	18	13

Pro rušení myčky (Tabulka 19) byla průměrná úspěšnost před filtrací pouze 2 %, kdy byly nejvíce rozeznány příkazy „Turn off the dishwasher“, který měla ale nulovou úspěšnost po filtraci. Průměrná úspěšnost pro algoritmus LMS i metodu ICA byla shodná a to 85 %.

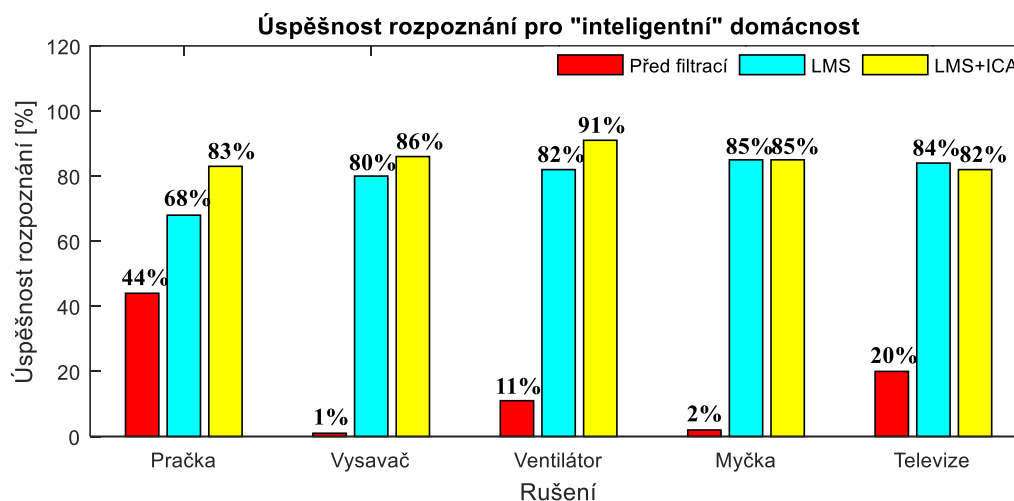
Tabulka 19 Výsledky úspěšnosti rozpoznání pro rušení myčky

LMS a ICA	Myčka maximální hlasitost		
	Před [%]	LMS [%]	LMS+ICA [%]
„light on“	0	100	100
„light off“	0	65	66
„blinds down“	0	100	100
„blinds up“	0	100	100
„blinds down left“	0	100	100
„blinds down right“	3	100	100
„blinds up left“	5	100	100
„blinds up right“	0	100	100
„Turn off the dishwasher“	10	0	0

Při zapnuté televizi (Tabulka 20) byla průměrná úspěšnost před filtrací pouze 20 %, kdy byly nejvíce rozeznány příkazy „blinds up middle“ 42 %, „blinds down left“ 24 %, „blinds down right“ 21 % a „blinds down middle“ 20 %. To je způsobeno tím, že takto dlouhá slova dokázal rozpoznávač zachytit mezi pauzami dialogu z televize. U algoritmu LMS byla průměrná úspěšnost 84 % a u metody ICA 82 %. Nejmenší úspěšnost rozpoznání po filtraci byla u příkazu „blinds down“, kdy rozpoznávač většinou vyhodnotil jinou alternativu („blinds down middle“, „blinds down right“, „blinds down left“)

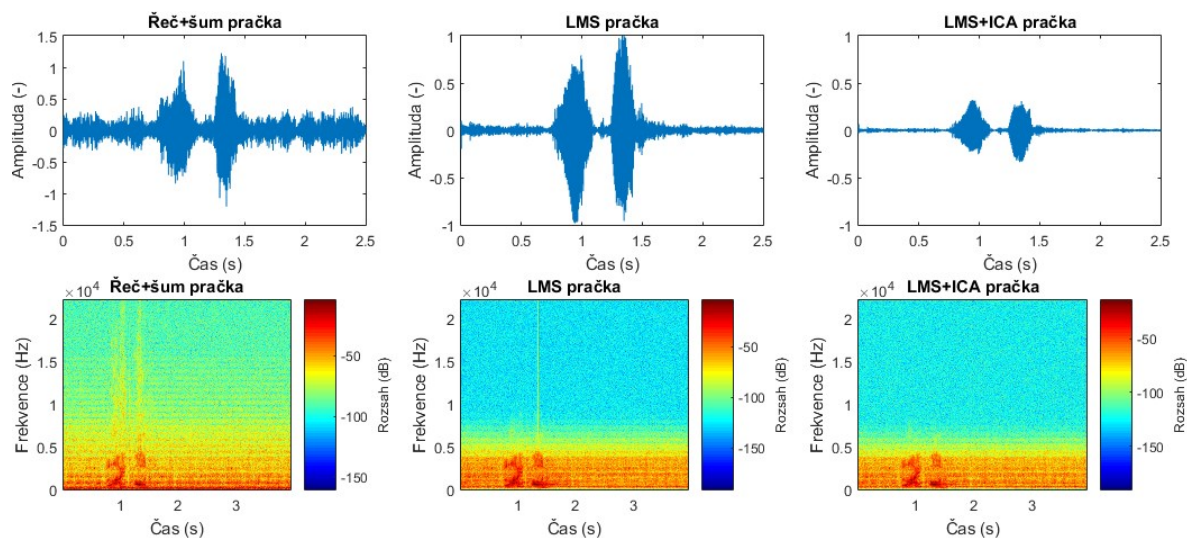
Tabulka 20 Výsledky úspěšnosti rozpoznání pro rušení televize

LMS a ICA	Televize maximální hlasitost		
Příkaz	Před [%]	LMS [%]	LMS+ICA [%]
„light on“	60	100	100
„light off“	0	74	68
„blinds down“	0	62	51
„blinds up“	0	74	60
„blinds down left“	24	97	80
„blinds down right“	21	98	96
„blinds down middle“	20	100	98
„blinds up left“	15	91	88
„blinds up right“	8	74	90
„blinds up middle“	42	100	98
„Turn off the TV“	27	52	69

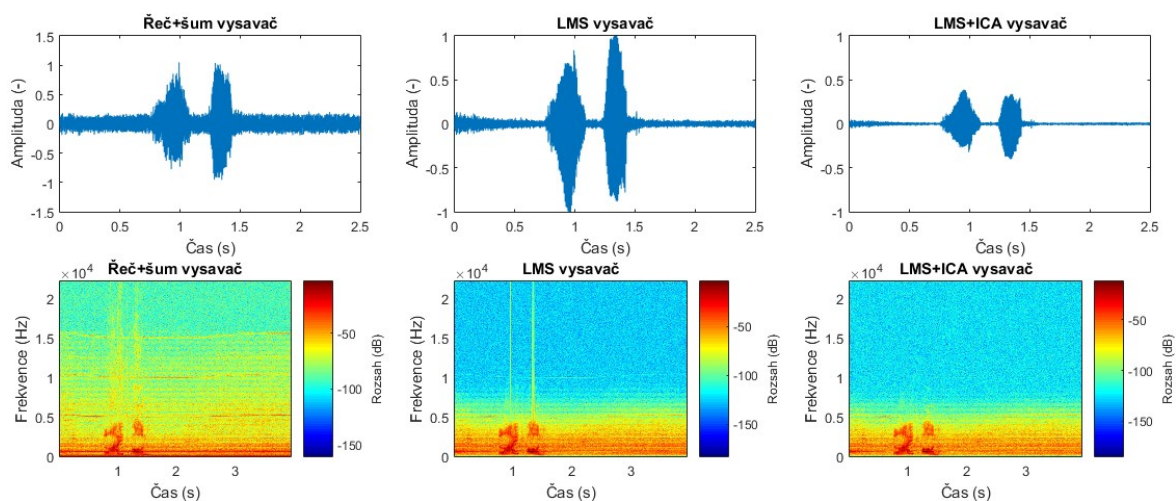


Obr. 77 Výsledky rozpoznání pro "inteligentní" domácnost

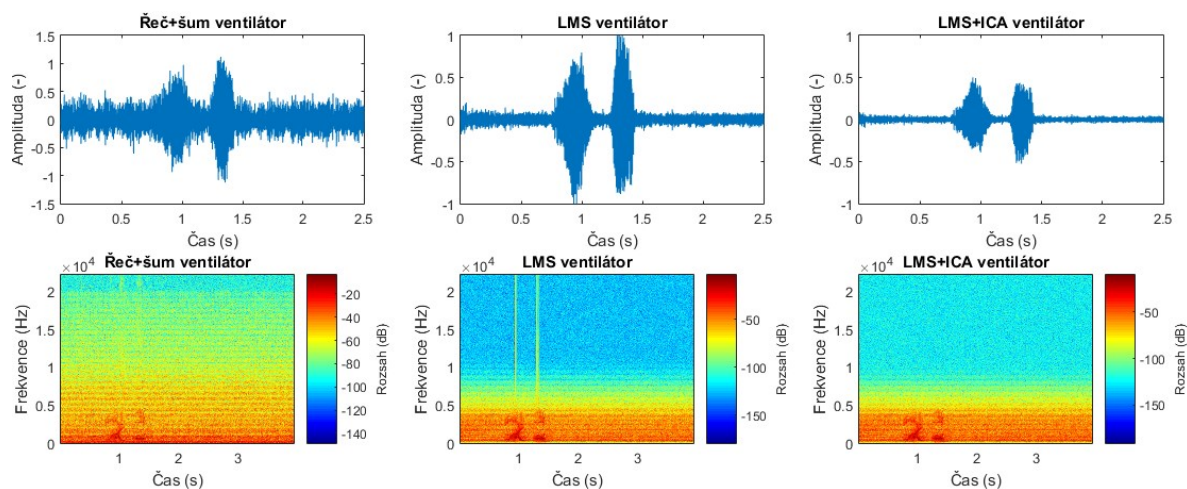
Ze spektrogramů (Obr. 78, Obr. 79, Obr. 80, Obr. 81, Obr. 82) lze vidět, že u spotřebičů je chování po filtraci podobné, zejména u pračky a vysavače. U ventilátoru jsou výsledky totožné. To je dáno především rovnoměrným rozložením šumu, který je blízký tomu Gaussovu. A z teorie adaptivní filtry s těmito šumy dobře nepracují a u metody ICA je to základní omezení, kdy nelze s těmito rušeními dobře pracovat. Ze spektrogramu dále vyplývá, že nejlépe odfiltrované rušení bylo u myčky. U rušení televize jde naopak vidět, že kvalita filtrace je na straně algoritmu LMS. Hlavní důvod je ten, že je využíván pouze jeden mikrofón, a tak nelze řešit klasický princip metody ICA.



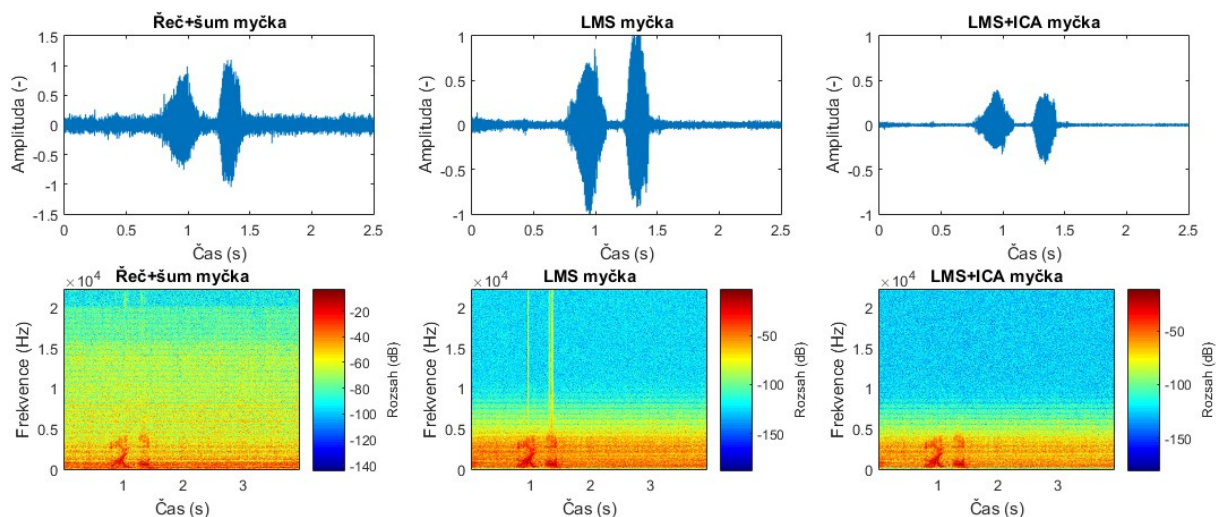
Obr. 78 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení pračky



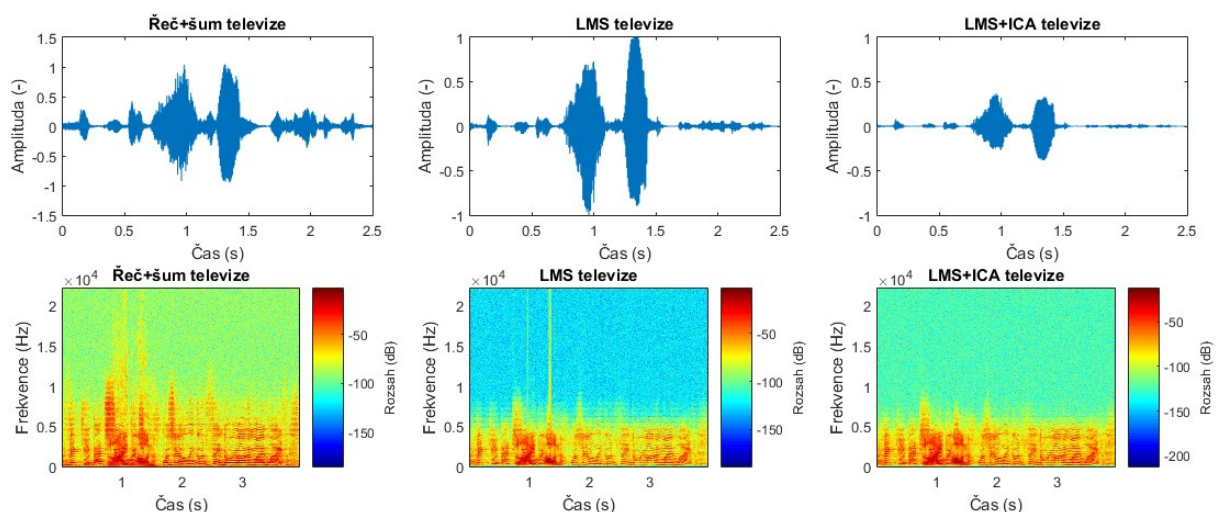
Obr. 79 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení vysavače



Obr. 80 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení ventilátoru



Obr. 81 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení myčky



Obr. 82 Porovnání spektrogramů při zavřených oknech, příkaz „light on“, rušení televize

7.4.3.3 Výrobní linka

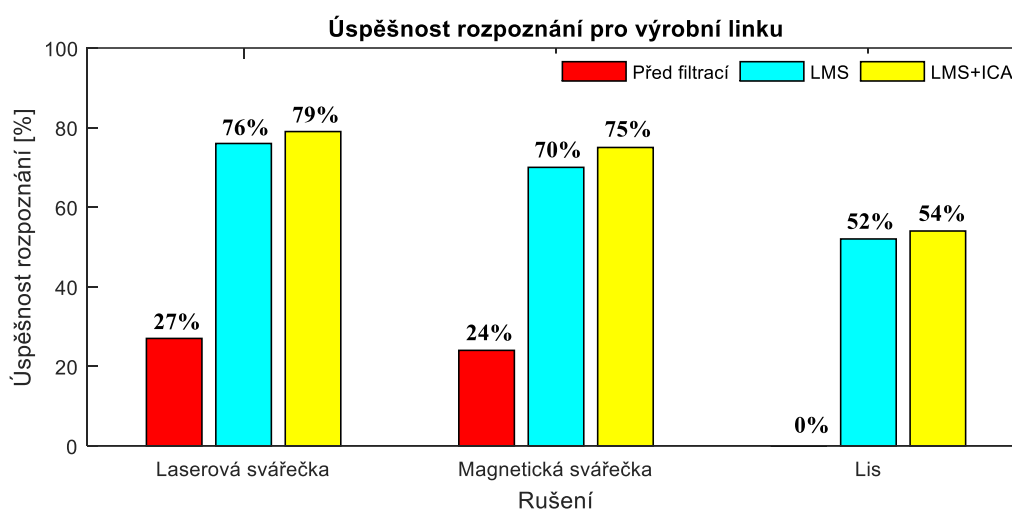
Z Tabulka 21 vyplývá, že při rušení laserové svářečky byla průměrná úspěšnost před filtrací 27 %, kdy nebyly rozpoznány příkazy „turn on the middle arm“, „turn on the left arm“. Po filtraci byla průměrná úspěšnost pro algoritmus LMS 76 % a pro metodu ICA 79 %.

Pro magnetickou svářečku byla průměrná úspěšnost před filtrací 24 %, kdy nejméně úspěšně rozpoznání bylo zaznamenáno u příkazu „turn on the left arm“. Po filtraci byla průměrná úspěšnost pro algoritmus LMS 70 % a pro metodu ICA 75 %. U obou rušení můžeme vidět, že u metody ICA je pro příkaz „turn off the laser welder“ přibližně 30 % zhoršení rozpoznání.

U rušení lisu lze vidět, že před filtrací je nulová úspěšnost. Energie rušení je tak velká, že bylo potřeba přiložit mikrofon těsně k ústům a zesílit tak užitečný signál. I přesto je průměrná úspěšnost po filtraci pro algoritmus LMS 52 % a pro metodu ICA 54 %. Podobně jako u měření automobilu při 130 km/h a otevřených oknech se jedná o krajní situaci, kdy byla testována robustnost obou metod.

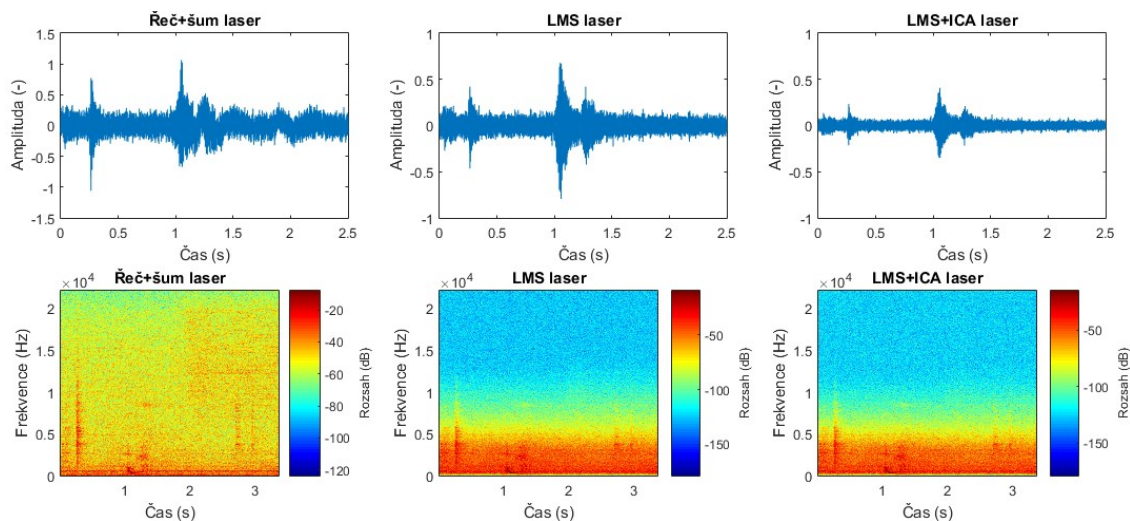
Tabulka 21 Výsledky úspěšnosti rozpoznání pro rušení laserové/magnetické svářečky a lisu

LMS a ICA	Výrobní linka								
	Laserová svářečka			Magnetická svářečka			Lis		
Příkaz	PŘED [%]	LMS [%]	LMS+ICA [%]	PŘED [%]	LMS [%]	LMS+ICA [%]	PŘED [%]	LMS [%]	LMS+ICA [%]
„Homing“	57	90	100	30	72	100	0	56	63
„Turn off the produc. line“	51	70	81	54	74	69	0	48	51
„Turn on the main arm“	18	85	87	18	63	72	0	45	39
„Turn on the right arm“	9	66	84	12	57	74	0	42	45
„Turn on the middle arm“	0	63	72	10	66	70	0	38	36
„Turn on the left arm“	0	66	81	3	66	78	0	36	45
„Turn off the laser welder“	21	100	70	18	100	69	0	100	100
„Turn on the belt“	60	69	57	50	62	71	0	51	50

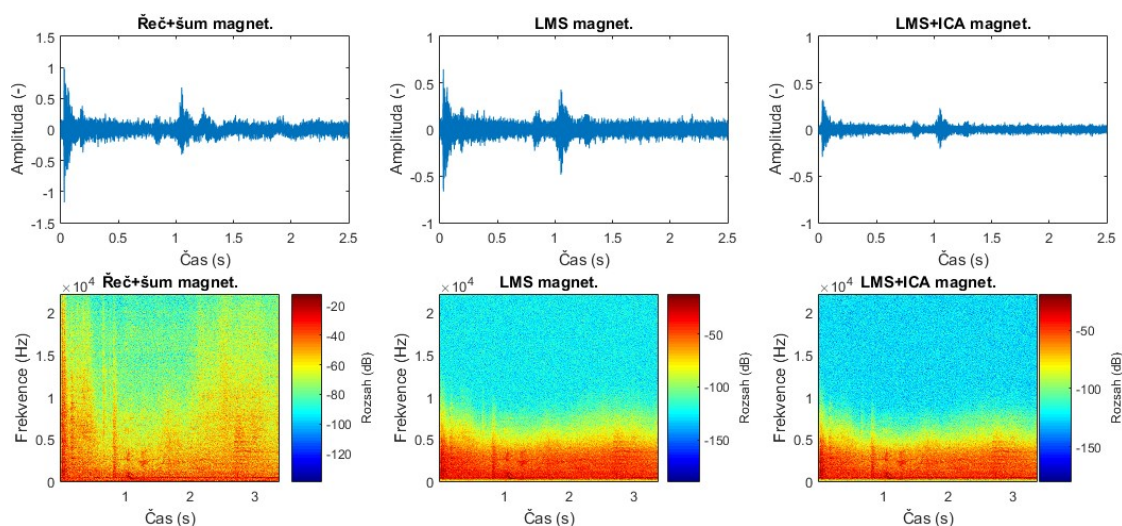


Obr. 83 Výsledky rozpoznání pro výrobní linku

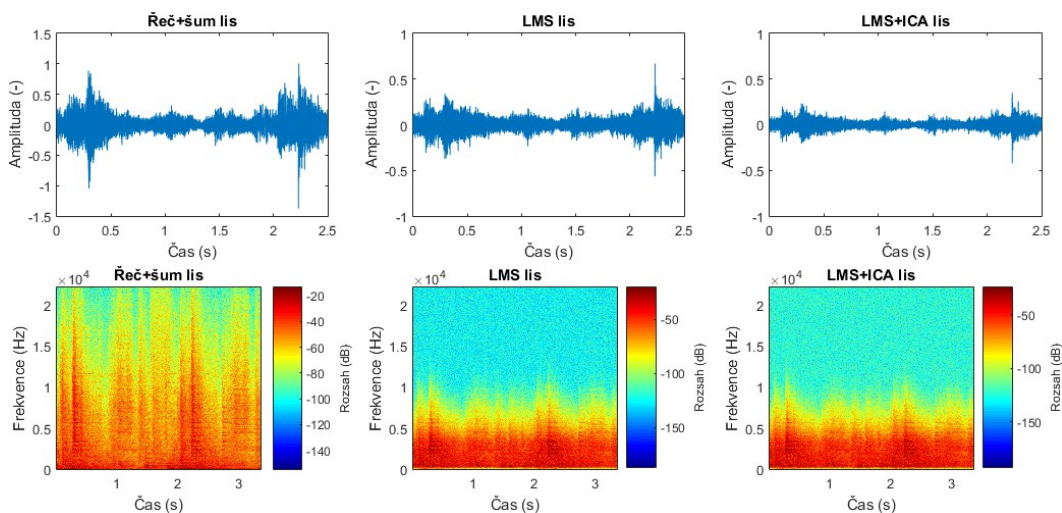
Ze spektrogramů (Obr. 84, Obr. 85, Obr. 86) lze vidět, že filtrované signály jsou u obou svářeček podobné, kdy kombinace LMS+ICA vykazuje mírné zlepšení filtrovaného signálu. To je dáno tím, že metoda ICA normalizuje signál na hodnotu rovnou polovině vstupního signálu, a tak redukuje vysoké hodnoty rušivého signálů. Můžeme tedy říct, že čím větší bude poměr šumu k řeči, tím se tato metoda bude jevit lepší.



Obr. 84 Porovnání spektrogramů, příkaz „homing“, rušení laserové svářečky



Obr. 85 Porovnání spektrogramů, příkaz „homing“, rušení magnetické svářečky



Obr. 86 Porovnání spektrogramů, příkaz „homing“, rušení lisu

Závěr

Tato diplomová práce je zaměřena na zpracování řečových signálů, konkrétně hlasové ovládání v oblasti Průmyslu 4.0 a SMART technologií pro následné testování vybraných filtračních metod. Teoretická část obsahuje informace o současných možnostech zpracování řečových signálů, metod automatického rozpoznávání řeči a metod pro potlačení rušení. Na závěr kapitol jsou jednotlivé metody sumarizovány a vyhodnoceny jejich přednosti a nedostatky.

Na základě prostudování dané problematiky byl pro filtraci řečových signálů, které byly kontaminovány šumem, vybrán zástupce lineárních adaptivních filtrů se stochasticky gradientní (rodina LMS) i rekurzivní optimální (rodina RLS) adaptací MSE a metoda nezávislých komponent ICA.

V realizaci byla nejprve navržena aplikace pro tvorbu databáze běžně vyskytujících se rušení. Na základě těchto nahrávek byly pro hlasové ovládání vytvořeny tři vizualizace: interiér automobilu, „inteligentní domácnost a výrobní linka.

Nejprve byly testovány adaptivní algoritmy typu LMS a RLS v offline režimu pro nalezení optimálních hodnot filtru. U algoritmu LMS byla zjištěna menší rychlost konvergence a vyšší chyba filtračního procesu. U algoritmu RLS byla zjištěna extrémně rychlá konvergence a přesnost na úkor matematické náročnosti, kde výpočetní čas trval v průměru 15 sekund. Takto vysoké časy zpracování jsou nepřijatelné pro systémy reálného času, a proto byl dále testován algoritmus LMS. Pro porovnání byla použita hybridní filtrace algoritmu LMS a metody ICA.

Při testování úspěšnosti rozpoznání byly použity nahrávky v kabině Škody Felicie, výrobní lince a polobezodrazové komoře. Kritériem pro úspěšnost rozpoznání byla sekvence sta opakování pro každý příkaz, na základě které byl vyhodnocen stav rozpoznal/nerozpoznal.

Z experimentu vyplývá, že při zavřených oknech při rychlostech 80 a 100 km/h je úspěšnost před filtrací 100 %. Při následném odfiltrování dochází k fenoménu známém především v telekomunikační technice, kdy filtrace hybridní metodu mírně zhoršila přesnost rozpoznávání na 92 %. Tento jev je znám například z metody RASTA nebo její modifikace RASTA-PLP. Pro kombinaci levého otevřeného okna vychází výsledky ve prospěch hybridní filtrace, a to přibližně o 5 %. Celkové zlepšení před a po filtraci je pro algoritmus LMS průměrně 45 % a pro hybridní filtraci 51,3 %. Pro otevřená okna je při 80 km/h rozdíl 12 % ve prospěch hybridní filtrace ale v rychlostech 100 a 130 km/h jsou rozdíly mezi 1-3 %. To je zapříčiněno především tlakovými pulzy způsobené poryvy větru kdy již filtry nejsou schopné dosáhnout lepších výsledků. Průměrná úspěšnost před a po filtraci je pro algoritmus LMS 36,6 % a pro hybridní filtraci je to 43,3 %.

Při testování „inteligentní“ domácnosti byly testovány příkazy pro pět druhů rušení. Z výsledků vyplývá, že hybridní metoda vykazovala vyšší úspěšnost rozpoznání než algoritmus LMS, a to v průměru o 6 %. Průměrná úspěšnost rozpoznání před a po filtraci byla pro algoritmus LMS o 64,2 % vyšší, pro hybridní filtraci o 69,8 %.

Posledním měřeným subjektem byla výrobní linka pro 3 druhy rušení. Z výsledků vyplývá, že hybridní metoda vykazovala vyšší úspěšnost rozpoznání než algoritmus LMS, a to v průměru o 3,3 %. Průměrná úspěšnost rozpoznání před a po filtraci byla pro algoritmus LMS o 49 % vyšší, pro hybridní filtraci o 52,3 %.

Z celkových výsledků je patrné, že hybridní filtrace vykazovala vyšší úspěšnost jen přibližně o 5 %. Pro výpočetní náročnost metody ICA je mnohem výhodnější implementovat algoritmus LMS, který je schopen vysoké míry filtrace i přes jeho jednoduchost, avšak při vzrůstajícím výkonu a kvalitě výpočetní techniky se otevírá prostor i pro komplikovanější algoritmy k řešení rozsáhlých úloh za relativně nízkou cenu.

Použitá literatura

- [1] AMRUTHA, ARAYND, ANSU MATHEW, SWATHY SUGATHAN, RAJASREE, PRIYALAKSHMI “Voice Controlled Smart Home”, IJETAE, Vol.5, Issue 1, January 2015.
- [2] HEM KAMDAR, ROSHAN KARKERA, ARCHIT KHANNA, PRANAV KULKARNI AND SUPRIYA AGRAWAL, “ A Review on Home Automation Using Voice Recognition,“ International Research Journal of Engineering and Technology (IRJET), Vol. 4, Issue 10, October 2017.
- [3] KANGO, R.; MOORE, R.; PU, J. “Networked smart home appliances-enabling real ubiquitous culture”, Proceedings of 5th International Workshop on Networked Appliances,Liverpool, 2002
- [4] WANG, Y. M.; RUSSELL, W.; ARORA, A.; JAGANNATHAN, R. K. XU, J. (2000). “Towards dependable home networking: an experience report”, Proceedings of the International Conference on Dependable Systems and Networks, p.43.
- [5] IAN MCLOUGHLIN AND HAMID REZA SHARIFZADEH, Speech Recognition for Smart Homes, chapter 27, Nanyang Technological University, Singapore, July 2008
- [6] RABINER, L. R,“Applications of voice processing to telecommunications”, In proceedings of the IEEE, Vol. 82, No. 2, pp. 199-228, February,1994
- [7] THORAYA OBAID, „Zigbee Based Voice Controlled Wireless Smart Home System“, International Journal of Wireless & Mobile Networks (IJWMN), Vol. 6, No.1, February 2014.
- [8] DHAWAN S. THAKUR AND ADITI SHARMA,Voice Recognition Wireless Home Automation System Based On Zigbee, IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), Volume 6, Issue 1 (May. -Jun. 2013), PP 65-75
- [9] MCTEAR, M. F, „Spoken Dialogue Technology: Toward The Conversational User Interface“, Springer Publications, 2004
- [10] CHEVALIER, H.; INGOLD, C.; KUNZ, C.; MOORE, C.; ROVEN, C.; YAMRON, J.; BAKER B.; BAMBERG, P.; BRIDLE, S.; BRUCE, T.; WEADER, A,“Large-vocabulary speech recognition in specialized domains”, Proc. ICASSP, Vol. 1 pp. 217-220, 1996
- [11] KAMM, C. A.; YANG, K.M.; SHAMIEH, C. R.; SINGHAL. S, “Speech recognition issues for directory assistance applications”, 2nd IEEE Workshop on Interactive Voice Technology for Telecommunications Applications IVTTA94, May, pp. 15-19, Kyoto, 1994

- [12] SUN, H.; SHUE, L.; CHEN, J., "Investigations into the relationship between measurable speech quality and speech recognition rate for telephony speech", Proc. ICASSP, May, Vol. 1, pp.1.865-1.868, 2004
- [13] Ravishankar, M. K., "Efficient algorithms for speech recognition", Ph.D thesis, Carnegie Mellon University, May, 1996
- [14] JAYASHRI VAJPAI a kol., Int. Journal of Engineering Research and Applications, ISSN: 2248-9622, Vol. 6, Issue 3, (Part - 1) March 2016, pp.88-95
- [15] Industrial robot statistics 2018, dostupné z <http://www.ifr.org/industrial-robots/statistics>
- [16] RAGOWSKI A., Robotics and Computer-Integrated Manufacturing Volume 28, Issue 3, Pages 303-315, June 2012
- [17] GABRIEL PIRES, URBANO NUNES, "A Wheelchair Steered through Voice Commands and Assisted by a Reactive Fuzzy-Logic Controller", Journal of Intelligent and Robotic Systems 34, 2002.
- [18] DOUGLAS W. BEEKS ROCKWELL COLLINS, "Speech Recognition and Synthesis", Chapter 8, the avionics handbook 2001 by CRC Press LLC
- [19] LAWRENCE R. RABINER, "Applications Of Speech Recognition In The Area Of Telecommunications"
IEEE, 1997 dostupné: <https://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwjU9ZG8qafLAhVQco4KHWnRBkYQFggcMAA&url=http%3A%2F%2Fwww.ece.ucsb.edu%2FFaculty%2FRabiner%2Fec259%2FReprints%2FApplications%2520of%2520speech%2520recognition%2520to%2520telecom.pdf&usg=AFQjCNFrLiTQTAtvlemflu3qEqSTBoD16g&sig2=VIRQgXqoiIfVc2wgnYMufQ>
- [20] MAŘÍK, VLADIMÍR. *Průmysl 4.0: výzva pro Českou republiku*. Praha: Management Press, 2016. ISBN 978-80-7261-440-0.
- [21] Cyber-Physical Systems. *Cisco Newsroom* [online].[cit.2018-11-11]. Dostupné z: <http://cyberphysicssystem.org/>
- [22] MARDIANA B., HAZURA H., FAUZIYAH S., ZAHARIAH M., HANIM A.R., NOOR SHAHIDA M.K., "Homes Appliances Controlled Using Speech Recognition in Wireless Network Environment," ICCTD International Conference on Computer Technology and Development, vol. 2, pp.285- 288, 2009
- [23] Smart Device Techopedia [online]. Dostupné z <https://www.techopedia.com/definition/31463/smart-device> [cit. 15. 1. 2019]

- [24] SCHIEFER, MICHAEL. Smart Home Definition and Security Threats. Ninth International Conference on IT Security Incident Management & IT Forensics, Magdeburg, 2015. ISBN 978-1-4799-9902-6.
- [25] KYAS, OTHMAR. How To Smart Home. Key Concept Press, 2013. ISBN 978-3-944980-00-3.
- [26] PSUTKA, Josef. Mluvíme s počítačem česky. Praha: Academia, 2006. ISBN 80-200-1309-1.
- [27] SAKOE, H. & S. CHIBA. Dynamic Programming Algorithm Optimization for Spoken Word Recognition. IEEE Transactions on Acoustics, Speech and Signal Processing 26, 1978, 43–49.
- [28] RICHARD E. BELLMAN and STUART E. DREYFUS. Applied dynamic programming. Princeton University Press, 1962
- [29] TOBOLÍKOVÁ P., Detekce chyb v rozpoznávání mluvené řeči, Univerzita Karlova v Praze, Matematicko-fyzikální fakulta, vedoucí práce: Jan Hajič
- [30] PAVEL IRCING (2017): ROZPOZNÁVÁNÍ ŘEČI. In: Petr Karlík, Marek Nekula, Jana Pleskalová (eds), CzechEncy-Nový encyklopedický slovník češtiny. URL: <https://www.czechency.org/slovník/ROZPOZNÁVÁNÍ ŘEČI> [cit. 21.3 2019]
- [31] ANKIT KUMAR, MOHIT DUA, “Continuous Hindi Speech Recognition using Monophone based Acoustic Modeling”, International Journal of Computer Applications , 0975 – 8887, 2014.
- [32] PREETI SAINI, PARNEET KAUR, “Automatic Speech Recognition: A Review”, International Journal of Engineering Trends and Technology-Volume 4 Issue2-2013.
- [33] SUMAN K. SAKSAMUDRE, SHRISHRIMAL P.P., DESHMUKH R.R., A Review on Different Approaches for Speech Recognition System, International Journal of Computer Applications, 0975 – 8887, Volume 115 – No. 22, April 2015
- [34] H. HERMANSKY, “Perceptual linear predictive (PLP) analysis of speech,” Acoustical Society of America Journal, vol. 87, pp.1738–1752, Apr. 1990.
- [35] LEI XIE, ZHI-QIANG LIU, “A Comparative Study of Audio Features For Audio to Visual Conversion in MPEG-4 Compliant Facial Animation,” Proc. of ICMLC, Dalian, 13-16 Aug-2006
- [36] A. GARG and P. SHARMA, "Survey on acoustic modeling and feature extraction for speech recognition," in Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on, 2016, pp. 2291-2295.

- [37] RAJNOHA. J., POLLÁK P., Detektory řečové aktivity na bázi perceptivní keprstrální analýzy. České vysoké učení technické v Praze, Fakulta elektrotechnická.
- [38] G. A. SAON and H. SOLTAU, "Method and system for joint training of hybrid neural networks for acoustic modeling in automatic speech recognition," ed: Google Patents, 2017.
- [39] GODINO-LLORENTE J.I., GOMEZ-VILDA P., and BLANCOVELASCO M., "Dimensionality Reduction of a pathological Voice Quality Assessment System Based on Gaussian Mixture Models and Short Term Cepstral Parameters", IEEE Transaction on Biomedical Engineering, 53(10):1943-1953, 2006.
- [40] LU-SHIH ALEX LOW, et al., "Content Based Clinical Depression Detection in Adolescents", 17th EUSIPCO 2009, Scotland Aug. 24-28, 2009.
- [41] L. H. LINH, N. T. HAI, N. VAN THUYEN, T. T. MAI, V. VAN TOI, MFCC-DTW algorithm for speech recognition in an intelligent wheelchair, in: Proceedings of the fifth International Conference on Biomedical Engineering in Vietnam, Springer International Publishing, Cham, 2015, pp. 417–421.
- [42] CHADAWAN ITTICHAICHAREON, SIWAT SUKSRI, THAWEESAK YINGTHAWORNSUK, Speech Recognition using MFCC, International Conference on Computer Graphics, Simulation and Modeling (ICGSM'2012) July 28-29, 2012 Pattaya (Thailand)
- [43] MATOUŠEK, J. Syntéza řeči z textu s využitím statistického přístupu k automatickému vytvoření databáze řečových jednotek: ARTIC – český TTS systém. Katedra Kybernetiky, ZČÚ, Plzeň, 2000. 164 stran.
- [44] VAŘÁK, JAN. Možnosti hlasového ovládání bezpilotních dronů [online]. Ostrava, 2017 [cit. 2019-02-28]. Dostupné z: <http://hdl.handle.net/10084/118887>. Bakalářská práce. Vysoká škola báňská - Technická univerzita Ostrava.
- [45] WOUTER GEVAERT, GEORGI TSENOV, VALERI MLADENOV, Neural Networks used for Speech Recognition, Journal of Automatic control, University of Belgrade, vol. 20:1-7, 2010
- [46] NOREZMI JAMAL, SHAHNOOR SHANTA, FARHANAHANI MAHMUD, MNAH SHA'ABANI, Automatic speech recognition (ASR) based approach for speech therapy of aphasic patients: A review, Conference: Advances in Electrical and Electronic Engineering: from theory to applications: Proceedings of the International Conference on Electrical and Electronic Engineering (IC3E 2017)

- [47] M. CUTAJAR, E. GATT, I. GRECH, O. CASHA, and J. MICALLEF, "Comparative study of automatic speech recognition techniques," *IET Signal Processing*, vol. 7, pp. 25-46, 2013.
- [48] S. XUE, O. ABDEL-HAMID, H. JIANG, L. DAI, AND Q. LIU, "Fast adaptation of deep neural network based on discriminant codes for speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, pp. 1713-1725, 2014.
- [49] S. ROMDHANI, "Implementation of DNN-HMM Acoustic Models for Phoneme Recognition," 2015.
- [50] P. PUJOL, S. POL, C. NADEU, A. HAGEN, AND H. BOURLARD, "Comparison and combination of features in a hybrid HMM/MLP and a HMM/GMM speech recognition system," *IEEE Transactions on Speech and Audio processing*, vol. 13, pp. 14-22, 2005.
- [51] E. ZARROUK, Y. B. AYED, AND F. GARGOURI, "Hybrid continuous speech recognition systems by HMM, MLP and SVM: a comparative study," *International Journal of Speech Technology*, vol. 17, pp. 223-233, 2014.
- [52] UPADHYAY, NAVNEET; KARMAKAR, ABHIJIT. Speech enhancement using spectral subtraction-type algorithms: A comparison and simulation Study. *Procedia Computer Science*, 2015, 54: 574-584.
- [53] SOUNDARYA, S.V.S a RAJANI ALUGONDA. A Review on Various Speech Enhancement Techniques. *International Journal of Advanced Research in Computer and Communication Engineering*. 2016, 5(8), 1-6. ISSN 2278-1021.
- [54] MARTINEK, RADEK. The Use of complex adaptive methods of signal processing for refining the diagnostic quality of the abdominal fetal cardiogram. Ostrava, 2014. Disertační práce. Vysoká škola báňská-Technická univerzita Ostrava. Vedoucí práce: Jan Židek.
- [55] JAN J. Číslíková filtrace, analýza a restaurace signálů. 2. vydání Vutium Brno 2002, 427 s. ISBN 80-214-2911-9.
- [56] HARDING, PHILIP JOHN. Model-Based Speech Enhancement. Norwich, 2013. Disertační práce. University of East Anglia. Vedoucí práce Dr. Ben Milner.
- [57] C. COLE, M. KARAM AND H. AGLAN, "Increasing Additive Noise Removal in Speech Processing Using Spectral Subtraction," *Fifth International Conference on Information Technology: New Generations (itng 2008)*, Las Vegas, NV, 2008, pp. 1146-1147. doi: 10.1109/ITNG.2008.86
- [58] LOIZOU, P.C. Speech enhancement: theory and practice. 2nd ed. Boca Raton, Fla.: CRC Press, 2013, 689 s. ISBN 978-1-4665-0421-9.

- [59] HOVORKA, J. Algoritmy spektrálního odečítání pro zvýraznění řeči v bojových pásových vozidlech. *Elektrorevue-Internetový časopis*. 2009, č. 35, s. 1-7. ISSN: 1213-1539.
- [60] RAJEEV A., JAI S. AND VIJAY G., “Noise Reduction of Speech Signal using Wavelet Transform with Modified Universal Threshold”, *International Journal of Computer Applications*, vol. 20, no. 5, 2011.
- [61] SLAVY G. MIHOV, RATCHO M. IVANOV, ANGEL N. POPOV, “Denoising Speech Signals by Wavelet Transform,” *Annual Journal Of Electronics*; 2009; ISSN 1313-1842.
- [62] MARTINEK, RADEK. Využití adaptivních algoritmů LMS a RLS v oblasti adaptivního potlačování šumu a rušení. 2013.
- [63] Adaptivní filtrace zašuměných řečových signálů. *Elektrorevue* [online]. Brno [cit. 2018-1-1]. Dostupné z: <http://www.elektrorevue.cz/clanky/02063/index.html>
- [64] FARHANG-BOROUJENY, B. Adaptive filters: theory and applications. 2nd ed. Chichester: Wiley, 2013. ISBN 978-1-119-97954-8.
- [65] SAEED V. VASEGHI. Advanced Digital Signal Processing and Noise Reduction. 3rd ed. Chichester: John Wiley Sons, 2006. ISBN 9780470094952.
- [66] MARTINEK, Radek; ZIDEK, Jan. The real implementation of NLMS channel ekvalizér into the system of software defined radio. *Advances in Electrical and Electronic Engineering*, 2012, 10.5: 330.
- [67] NATIONAL INSTRUMENTS CORP. Manuals–LabVIEW 2016 Adaptive Filter Toolkit Help. National Instruments Corp, 2016.
- [68] ERIK VISSER, MANABU OTSUKA, TE-WON-LEE, A spatio-temporal speech enhancement scheme for robust speech recognition in noisy environments, *Speech Communication*, Volume 41, Issues 2–3, October 2003, Pages 393-407
- [69] ERIK VISSER, TE-WON-LEE, Speech enhancement using blind source separation and two-channel energy based speaker detection, Institute for Neural Computation, University of California, San Diego, La Jolla, CA 92093-0523
- [71] HYVÄRINEN, A., OJA, A. Fast Fixed-Point Algorithm for Independent Component Analysis. *Neural Computation*. 1997, vol. 9, no. 7, p. 1483-1492. ISSN 0899-7667.
- [72] CICHOCKI, A., AMARI, S., Adaptive Blind Signal and Image processing, Japan, John Wiley & Sons, ISBN 0471 60791 6, 2002

- [73] ŠEBESTA, V. Teorie sdělování. Skripta, VUT v Brně, 1998. ISBN 80-214-1247-X
- [74] WEISSTEIN, E., W, Singular Value Decomposition From MathWorld – A Wolfram Web Resource. Dostupné z <<http://mathworld.wolfram.com/SingularValueDecomposition.html>>,[cit. 2. 2 2019]
- [75] FISCHER S ,SIMMER KU. Beamforming microphone arrays for speech acquisition in noisy environments. Speech communication, 20:215–227, 1996.
- [76] GRIFFITHS LJ, CHARLES WJ. An alternative approach to linearly constrained adaptive beamforming. IEEE Transaction on Antenas and Propagation, AP-30(1), January 1982.
- [77] QIYUE ZOU, ZHU LIANG YU, ZHIPING LIN, A Robust Algorithm for Linearly Constrained Adaptive Beamforming, IEEE signal processing letters, Vol. 11, no. 1, January 2004
- [78] PŘIKRYL, PETR. Porovnání úspěšnosti vícekanálových metod separace řečových signálů [online]. Brno, 2008 [cit. 28.2]. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=8569.Diplomová práce. Vysoké učení technice v Brně.
- [79] https://download.steinberg.net/downloads_hardware/UR44/UR44_documentation/UR44OperationManual_en.pdf
- [80] https://kytary.cz/steinbergur44/HN145128/?gclid=Cj0KCQjw4qvlBRDiARIsAHme6ouvg3MNUvyTG_Buwj85FPJveYwuF4QAwzraZTwbRu_1waPr6tM9v0UaAmEKEALw_wcB
- [81] <https://www.bswusa.com/Condenser-Microphones-Rode-NT5-Matched-Pair-P4635.aspx>
- [82] WITTASEK, Tomáš. Virtuální Instrumentace I. Ostrava: VŠB – Technická univerzita, 2012
- [83] <https://www.zeitnitz.eu/scms/waveio?mid=10>
- [84] ČERNOCKÝ J., Zpracování řečových signálů — studijní opora, Ústav počítačové grafiky a multimédií Fakulta informačních technologií, Vysoké učení technické v Brně. Dostupné z: http://www.fit.vutbr.cz/study/courses/ZRE/public/opora/zre_opora.pdf [cit. 10. 2. 2019]

Příloha A

Tabulka 22 Obsah přiloženého CD disku

Název adresáře	Popis
LabVIEW programy	Kódy realizovaných aplikací
Ovladače	Ovladače nutné k provozu vizualizací
Databáze nahrávek	Databáze vytvořených nahrávek řeči a rušení
Text	Diplomová práce v elektronické podobě